2 1977

P. 1877/77
POLITECHNIKA ŚLĄSKA BIBLIOTEKA GŁÓWNA w Gliwicach

informatyka

SPIS TREŚCI NUMERU 2/77

str.

Sieci komputerowe
Mieczysław Bazewicz 1

Język „MINI" jako propozycja w zakresie nauczania podstaw informatyki
Ryszard Tadeusiewicz 5

Realizacja systemu APO-3
Krzysztof Marciniak, Barbara Putz 9

Rozkład instrukcji w programach systemów informatycznych
Jerzy Leyk 12

Adaptacja pakietów TEXTPAC i STAIRS w informatycznych systemach dokumentacyjnych
Anna Pastusiak, Natalia Peczyńska, Teresa Rumińska 15

Założenia i rola podsystemu „INFRASTRUKTURA" w informatycznym systemie sterowania inwestycjami miejskimi
Wojciech Pietraszewski 19

Z KRAJU

III Krajowy Przegląd Zastosowań Techniki Komputerowej w Przemyśle Maszynowym. Poznań 19—21.X.1976 r.
Andrzej Klimek 23

Uwagi i spostrzeżenia uczestnika konferencji w Poznaniu
Andrzej Sokołowski 24

X-lecie ośrodka zrodzonego z potrzeb i entuzjazmu
Krystyn Bernatowicz 25

MERA-303 w szczecińskiej szkole *(P)* 26

Międzynarodowa Konferencja: Zastosowanie informatyki w przedsiębiorstwach przemysłu maszynowego COMPCONTROL-77 27

ZE ŚWIATA

Tydzień Techniki Bułgarskiej w Polsce
Andrzej Klimek 28

Kartka z Finlandii
oprac. *Antoni Wolski* 28

Informatyka po angielsku. Nowoczesność w gospodarce zapasami
Janusz Ilczuk 29

TERMINOLOGIA

Krajowa konferencja maszyn matematycznych w USA *(AEK)* 8

Czy język ojczysty będzie językiem informatyków
Włodzimierz Rabij 31

O TEZAURUSIE
Krystyna Siwek 32

PROBLEMATYKA TRANSMISJI DANYCH

Sieci transmisji danych — problematyka organizacji i funkcjonowania
Jerzy Kurkiewicz, Krzysztof Odolak, Wojciech Zołotucho 33

PRZEGLĄD WYDAWNICTW

Pomiędzy człowiekiem a maszyną cyfrową
Marek Hołyński 35

Bibliografia wydawnictw polskich z dziedziny informatyki
oprac. *A. K.* 14 i 22

Współczesne kryteria oceny drukarek *(WK)* III okł.

Ogłoszenie IV okł.

WYDAWNICTWA CZASOPISM TECHNICZNYCH NOT
Warszawa
Czackiego 3/5

KOLEGIUM REDAKCYJNE

Redaktor Naczelny prof. dr hab. Leon ŁUKASZEWICZ
mgr Krystyn BERNATOWICZ, prof. dr hab. inż. Konrad FIAŁKOWSKI (zast. redaktora naczelnego), dr inż. Marek HOŁYŃSKI, mgr inż. Stanisław JASKÓLSKI, Władysław KLEPACZ (zast. redaktora naczelnego), mgr Andrzej KLIMEK, doc. dr inż. Andrzej TARGOWSKI
Red. techn. Anna POTARSKA

RADA PROGRAMOWA

Mgr inż. Andrzej BOSSOWSKI, doc. dr inż. Jan FELICKI, doc. dr inż. Zbigniew GACKOWSKI, doc. dr inż. Aleksander GOLINOWSKI, dr hab. Andrzej KRZYWAK, doc. dr hab. inż. Roman KULESZA, prof. dr hab. Leon ŁUKASZEWICZ, prof. dr hab. inż. Stanisław PASZKOWSKI (wiceprzewodniczący), prof. dr Tadeusz PECHE, mgr inż. Bronisław PIWOWAR, dr inż. Andrzej PLĄSKOWSKI, mgr inż. Tadeusz PODGÓRSKI (wiceprzewodniczący), prof. dr inż. Jerzy SEIDLER, prof. dr inż. Andrzej STRASZAK (przewodniczący), doc. Jerzy TRYbulski, dr Tadeusz WALCZAK, prof. dr inż. Stefan WĘGRZYN, dr inż. Jan Z. ŻYDOWO

Redakcja: 60-041 Warszawa, ul. Jasna 14/16, pokój 331, tel. 27-71-40 lub centrala 26-82-61 w. 285, dyżury redakcji 10.00—13.00

Zakład Kolportażu WCT NOT, Warszawa, ul. Mazowiecka 12

Zakł. Graf. „Tamka". Z. 2. Zam. 12/701. Popier druk. sat. III kl. 80 g. A1, obj. 4,5 ark. druk. Nakład 7100. F-87.

Cena egzemplarza zł 15.— INDEKS 36210/36124 Prenumerata roczna zł 180.—

Informatyka

zastosowania w gospodarce, technice i nauce

Nr 2
MIESIĘCZNIK
1977
ROK XII
luty

P. 1877/77

ORGAN MINISTERSTWA NAUKI, SZKOLNICTWA WYŻSZEGO I TECHNIKI I POLSKIEGO KOMITETU AUTOMATYCZNEGO PRZETWARZANIA INFORMACJI NACZELNEJ ORGANIZACJI TECHNICZNEJ

MIECZYSŁAW BAZEWICZ
Politechnika Wrocławska

Sieci komputerowe

Rozwój współczesnych środków i metod informatyki polega przede wszystkim na doskonaleniu elementów i struktur złożonych systemów cyfrowych oraz udostępnianiu jak największej liczbie użytkowników dynamicznie wzrastających zasobów systemów cyfrowych. Wyrazem tych tendencji są systemy wielodostępne, sieci komputerowe, rozproszone bazy danych oraz rozwój interakcyjnych metod w natychmiastowej obsłudze dużej liczby użytkowników komputerów.

Kształtowanie się pojęć dotyczących sieci komputerowych dokonywało się wraz z postępem w rozpoznawaniu własności, specyfiki i form użytkowania coraz to nowych rozwiązań sprzętowo-programowych. W rozwiązaniach wielodostępnych i sieciowych koncentrowano więc kolejno uwagę na sprzęcie oraz na formach użytkowania i zastosowań. W związku z tym pojawiły się różne określenia i definicje systemu wielodostępnego i sieci komputerowej, takie jak: systemy abonenckie (Time Sharing Systems), sieci transmisji danych (Data Communication Systems), sieci teleinformatyczne (Telecommunication and Information Networks), systemy rozproszonych baz danych (Distributed Data Bases), sieci komputerowe (Computer Networks). Sieci komputerowe powstały w wyniku uogólnienia funkcji systemów informacyjnych, a także w wyniku przeobrażeń środków informatyki służących przesyłaniu informacji.

Przechodzenie w rozwoju sieci od form eksperymentalnych do rozwiązań utylitarnych sprawia, że coraz lepiej znane stają się ich funkcje wewnętrzne i zewnętrzne. Pod tym względem poszukuje się analogii między sieciami komputerowymi a innymi: telefonicznymi, pocztowymi, energetycznymi. Jednak ze względu na charakter zasobów występujących w sieci komputerowej najbliższa wydaje się ich analogia z systemami energetycznymi.

Najogólniej przyjmuje się, że sieć komputerową tworzą:

— co najmniej dwa systemy cyfrowe z podziałem czasu i wyposażeniem komunikacyjnym

— urządzenia abonenckie

— zespół technicznych środków telekomunikacji (komputery komunikacyjne, modemy, linie łączności itp.)

— oprogramowanie podstawowe sieci (sterowanie przesyłaniem, komunikacją i wymianą komunikatów między abonentami i komputerami w sieci).

Całość elementów i oprogramowania realizująca funkcje obsługi rozproszonych przestrzennie użytkowników jest nazywana zbiorem zasobów sieci komputerowej. Zbiór reguł dotyczących zasad współpracy elementów oraz abonentów w sieci jest nazywany protokółem sieci. Zasoby sieci dzieli się na fizyczne (sprzęt) i logiczne (oprogramowanie i zbiory danych).

Im bardziej zróżnicowane są pod względem walorów użytkowych zasoby sieci, tym większą satysfakcję sprawia jej użytkowanie.

Odnosi się to zarówno do użytkowników systemu wielodostępnego, jak i terminali włączonych bezpośrednio do sieci za pośrednictwem koncentratorów.

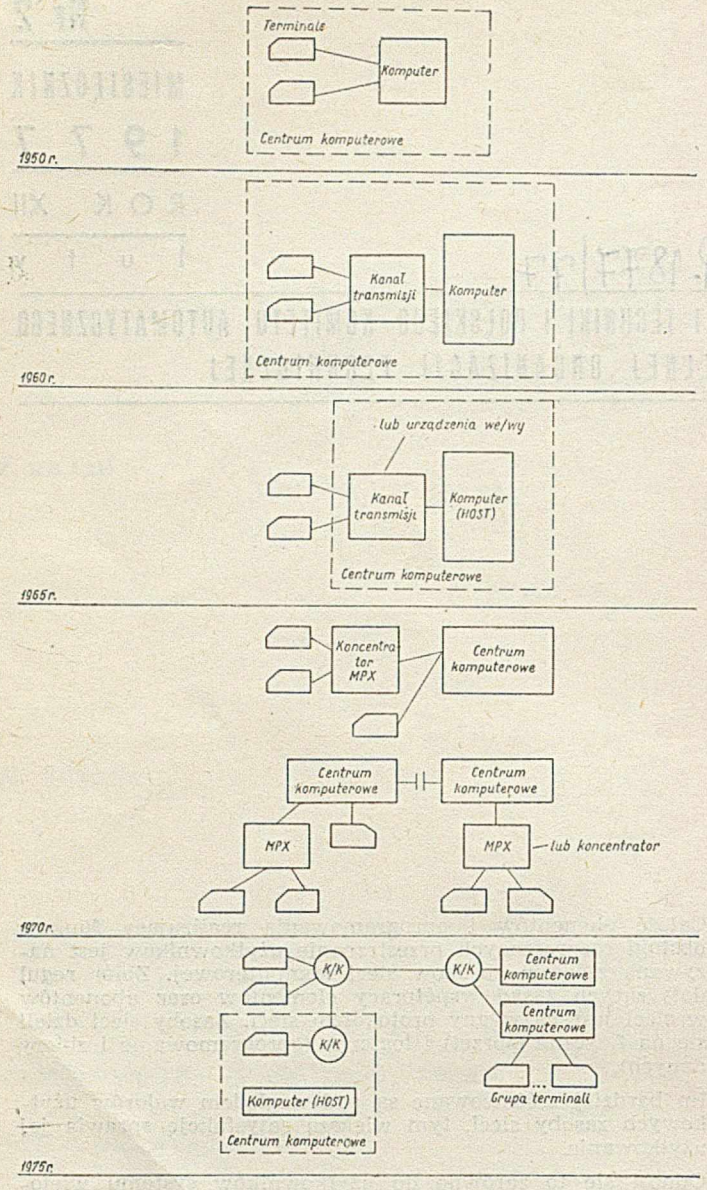
Ten punkt widzenia umożliwia określenie różnicy między systemem wielodostępnym a siecią komputerową. Wielodostęp polega na rozszerzeniu równoczesnego dostępu do zasobów systemu cyfrowego, sieć natomiast wzbogaca użytkownikowi zbiór tych zasobów.

STAN I ROZWÓJ SIECI

Początki rozwoju systemów komputerowych wykorzystujących środki transmisji danych sięgają lat pięćdziesiątych (rys. 1). Zasadnicze jednak przeobrażenia nastąpiły dopiero w latach sześćdziesiątych. Pierwsze rozwiązania polegały na bezpośredniej lokalnej współpracy terminala z procesorem. Następstwem tego było pojawienie się kanału komunikacyjnego dla lokalnej pracy kilku terminali z procesorem.

Linie komunikacji do pracy zdalnej oraz procesor we/wy powstały na początku lat sześćdziesiątych. W rezultacie tego w obsłudze zdalnej znalazły zastosowanie koncentratory i multipleksory.

W ostatnim dziesięcioleciu stosuje się procesory komunikacyjne oraz złożone systemy cyfrowe do współpracy komputerów i dużych ośrodków komputerowych we wspólnej sieci. Złożoność struktur systemów i ich funkcji użytkowych zapoczątkowała w latach sześćdziesiątych szybki rozwój rozwiązań programowych sterowania pracą sieci i obsługi użytkowników. Różnorodność systemów komputerowych, stosowanych standardów i zastosowania sieci uniemożliwia obecnie stworzenie jednoznacznej metodologii sieci komputerowej. Niemniej jednak rozwija się w tym kierunku intensywne prace, które, na razie — doprowadziły



Rys. 1. Dotychczasowe przeobrażenia w konfiguracjach sprzętu sieci

do sformułowań metodologicznych mniej uniwersalnych, ograniczonych do jednego typu systemu komputerowego.

Przykładem może być ramowa koncepcja metodologiczna stworzona na bazie rozwiązań systemowych komputerów IBM pod nazwą Systems Network Architecture (SNA). Wyrazem zwiększającego się zainteresowania tą dziedziną jest nowe czasopismo (od 1976 r.) pt. „Computer Networks — The International Journal of Distributed Informatique" o zasięgu światowym.

Nie ma na ogół różnicy zdań co do tego, że kierunki dotychczasowego rozwoju systemów wielodostępnych i sieci komputerowych zostały zdeterminowane potrzebami ośrodków uniwersyteckich i wojskowych. Potrzeba wykorzystania własności komputerów, które umożliwiają wprowadzenie do nauczania i badań najbardziej nowoczesnych i twórczych elementów spowodowały znaczną koncentrację wysiłków badawczych. Podjęte początkowo na gruncie eksperymentalnym — poszukiwania rozwiązań sieciowych przyczyniły się do uzyskania wyników o charakterze użytkowym, których przykładem mogą być sieci CTSS, ARPA, OCTOPUS, TYMNET, MERIT (w USA), AIST (w ZSRR) oraz CYKLADES (we Francji).

Pierwszymi sieciami komputerowymi były sieci typu interakcyjnego (pytanie — odpowiedź). Jednolity sposób obsługi zgłoszeń polegający na dostępie do danych i programów nie dawał możliwości ich modyfikacji przez bezpośredniego użytkownika. Ten rodzaj sieci ciągle zmieniano i udoskonalano. Dzięki temu jest on obecnie najpowszechniej stosowany w systemach rezerwacji miejsc, obsługi banków, w lotnictwie itp.

Kolejnym etapem w rozszerzaniu zastosowań sieci było stworzenie możliwości zdalnego dostępu do zasobów systemu oraz ich doskonalenie. Potrzeba obsługi większej liczby terminali przy ustalonym czasie reakcji, udostępnienie wspólnej bazy danych odległym użytkownikom przyczyniły się do osiągnięcia rozwiązań umożliwiających alternatywny dobór sprzętu, wyższą niezawodność sieci oraz mniejsze koszty transmisji. W konsekwencji — dzięki specjalizacji poszczególnych ośrodków wchodzących w skład sieci — użytkownicy uzyskali większy asortyment usług.

Obecnie, ze względu na występowanie w sieci dużych mocy obliczeniowych i jednocześnie potrzebę wyrównania obciążenia poszczególnych centrów obliczeniowych w celu uzyskania większej ich wydajności i niezawodności, zmierza się w kierunku rozwiązań umożliwiających współużytkowanie zasobów. Rozwiązania te wynikają w mniejszym stopniu z potrzeb użytkowników, w większym natomiast z konieczności racjonalnego wykorzystania sprzętu, polegają zaś m.in. na podziale zadań obliczeniowych między wyspecjalizowane ośrodki oraz na grupowaniu urządzeń w celu udostępnienia ich rozproszonym przestrzennie różnym typom komputerów. Przykładem sieci, w których występują — częściowo lub całościowo — wymienione rozwiązania są sieci DCS i CYBERNET eksploatowane w Stanach Zjednoczonych.

Ponieważ dotychczasowy rozwój sieci związany jest raczej z ich specjalizacją w określonych rodzajach zastosowań, przewiduje się, że dalszy rozwój będzie zmierzał w kierunku budowy sieci powszechnego użytku. Wymagane są więc sieci komutowane o charakterze publicznym, wykorzystujące cyfrowe techniki transmisji oraz wielodostępne systemy komputerowe.

ELEMENTY SIECI

W każdej sieci komputerowej wyróżnić można trzy podstawowe elementy: komputery użytkowników sieci (HOST), węzły komutacyjne i urządzenia telekomunikacyjne. Węzły i urządzenia telekomunikacyjne stanowią razem podsieć komunikacyjną.

Komputerem użytkowników sieci (HOST) jest zazwyczaj system wielodostępny, wykorzystywany w trybie wsadowym i interakcyjnym zarówno lokalnie jak i zdalnie.

Wyposażony on jest w mechanizmy podziału czasu oraz moduł programowy umożliwiający jego współpracę z komputerem komunikacyjnym. W niektórych przypadkach może być stosowany koncentrator, który umożliwia użytkownikom dostęp do sieci. Czasem koncentrator realizuje jednocześnie funkcje węzła komunikacyjnego.

Węzeł komutacyjny umożliwia kojarzenie abonentów sieci i steruje przepływem informacji między nimi. Funkcje węzła pełni zazwyczaj procesor zwany komputerem komunikacyjnym. Czasem może to być procesor, który równocześnie realizuje funkcje dostępu abonenta do sieci typu FEP (ang. Front End Procesor) lub rozbudowany terminal programowy.

Do urządzeń telekomunikacyjnych zalicza się modemy, centrale komutacyjne, urządzenia końcowe linii oraz linie transmisji.

Całość tych urządzeń stanowi łącze telekomunikacyjne, które może być tworzone na bazie:
— telefonicznej lub telegraficznej komutacyjnej sieci powszechnego użytku
— telefonicznych lub telegraficznych liniach dzierżawionych
— specjalnych, szerokopasmowych kanałów teletransmisyjnych.

O wyborze rodzaju łącza decydują: relacje współpracy urządzeń, wymagana jakość i szybkość łącza oraz koszty transmisji. Łącze charakteryzują parametry szybkości transmisji i stopy błędów, których wartość decyduje o przydatności łącza dla sieci komputerowej. Najlepsze pod względem jakości są łącza szerokopasmowe, jednak ich wysoki koszt sprawia, że aktualnie najszerzej stosuje się łącza dzierżawione.

Znane są dwa rodzaje techniki transmisji danych w sieciach: analogowa i cyfrowa (tzw. PCM) oraz synchroniczna i asynchroniczna. Technikę analogową można stosować dla

szybkości do 10 kbodów. Przyszłość transmisji o dużych szybkościach (od 20 kbodów do 50 kbodów) wiąże się natomiast z zastosowaniem techniki cyfrowej, a w dalszej perspektywie — łączności świetlnej i laserów (10^{10} bodów).

Ze względu na różnice czasu w transmisji danych i przetwarzaniu stosuje się dwie metody współpracy procesora z linią transmisji.

Metody te polegają na grupowaniu bitów z linii w znaki lub słowa (buforowanie znaku) oraz na kompletowaniu ciągu znaków czy też słów w komunikaty (buforowanie komunikatu). Ta ostatnia metoda jest bardziej efektywna, ponieważ odciąża procesor główny od obsługi transmisji. Wymaga ona stosowania dodatkowych urządzeń (procesor komunikacyjny), wskutek czego jest bardziej kosztowna.

Ważnym zagadnieniem w transmisji jest technika komutacji.

Obecnie stosowane są komutacje łączy i informacji. Pierwsza związana jest z komutacją kanałów tworzących bezpośrednie połączenia komputerów (lub abonentów) włączonych do sieci. Z punktu widzenia realizacji technicznej może to być komutacja łączy przestrzenna lub czasowa. Komutacja informacji polega na tym, że przysyłaną informację przechowuje się w pamięci węzła komutacyjnego sieci (w komputerze komunikacyjnym), skąd następnie kieruje się ją blokami do miejsca przeznaczenia.

Rozróżnia się formy przesyłania bloków informacji o zmiennej długości, zwanej komutacją komunikatów oraz stałej długości, zwanej komutacją pakietów. W komutacji pakietowej informacja jest dzielona w węzłach na bloki—pakiety i powtórnie składana w węźle przeznaczenia.

Mimo, że ta forma jest bardziej złożona i kosztowniejsza, to jednak przyszłość sieci jest związana z jej rozwojem.

W topologii struktury sieci znane są trzy rodzaje organizacji elementów i konfiguracji sprzętu, zwane scentralizowaną (gwiaździstą), rozłożoną (o różnym stopniu wzajemnych połączeń) i pętlową (łańcuchową).

Niezależnie od wymienionych skutków występują także rozwiązania pośrednie o charakterze mieszanym.

OPROGRAMOWANIE SIECI KOMPUTEROWYCH

Całość oprogramowania sieci, nazywana oprogramowaniem komunikacyjnym sieci, można podzielić na oprogramowanie komputera użytkownika (HOST) i oprogramowanie podsieci komunikacyjnej.

Oprogramowanie komunikacyjne sieci ma strukturę hierarchiczną, czteropoziomową.

Najniższy poziom, odpowiadający transmisji węzeł — węzeł, stanowią programy operacyjne komputerów komunikacyjnych. Realizują one funkcje przesyłania i analizy informacji sterujących transmisją, sprawdzania poprawności transmisji oraz inne zadania związane z tzw. protokółem sieci. Programy te prowadzą także statystykę przepływu pakietów oraz poprawności sieci i na tej podstawie umożliwiają wybór optymalnej drogi przesyłania pakietów w sieci.

Poziom drugi, odpowiadający transmisji węzeł — komputer użytkownika, stanowią programy sieciowe systemu operacyjnego komputera użytkownika sieci, na ogół wielodostępnego. Programy te są wywoływane przerwaniami generowanymi z komputerów komunikacyjnych, a ich zadaniem jest realizacja komunikacji węzła z komputerem użytkownika.

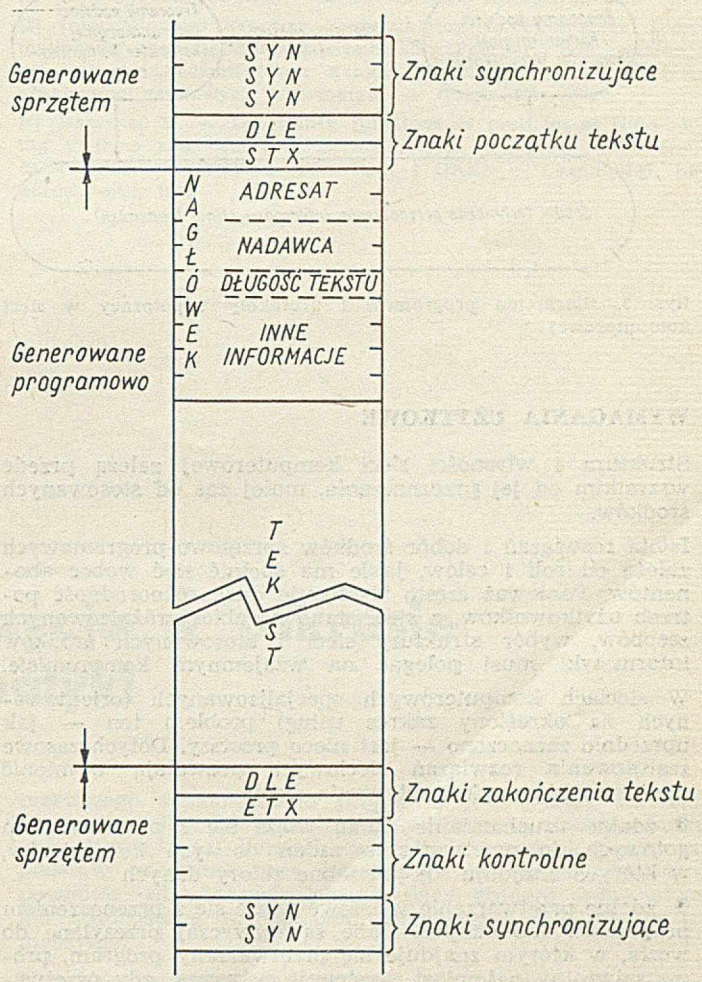
Poziom trzeci, dotyczący transmisji: komputer użytkownika — komputer użytkownika (HOST — HOST) w sieci, stanowi oprogramowanie realizujące funkcje systemowe.

Funkcje te dotyczą planowania i sterowania przesyłaniem zadań do innego komputera w przypadku wymiany obciążeń w sieci, organizacji współpracy między procesami przetwarzania w dwóch (lub więcej) komputerach użytkownika sieci oraz prowadzenia protokółu współpracy międzyprocesowej, planowania dostępu oraz organizacji współpracy z bazami danych różnych komputerów sieci.

Transmisja odpowiadająca czwartemu poziomowi abonent — abonent, stanowi oprogramowanie zapewniające realizację wymaganych przez użytkownika usług sieci. Programy te realizują protokół współpracy abonentów sieci i wiążą się bardzo ściśle z wymaganiami użytkowymi abonentów: z zastosowaniami sieci i formami współpracy użytkownika z siecią.

Wymiana informacji między komputerami użytkowników dokonywana jest poprzez węzły za pomocą specjalnych procedur. Procedury te stanowią moduł programowy komputera. Wysyłane informacje stanowią formę listów zawierających pytanie bądź program do realizacji, albo odpowiedź na pytanie, czy też wyniki przetworzonego programu.

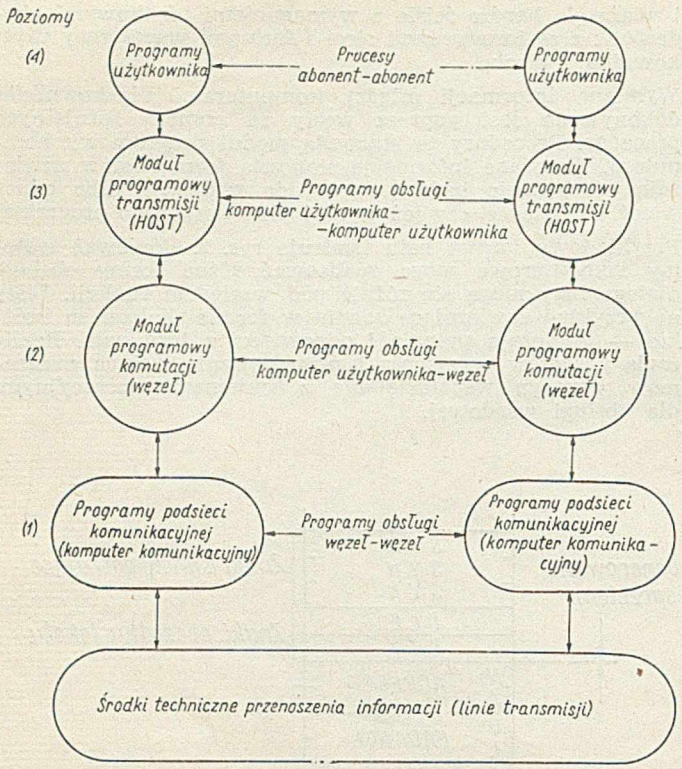
Przykładowy format listu ilustruje rys. 2. Ponieważ systemy komputerowe mogą realizować różne formy obsługi, dlatego listy mogą się różnić pod względem funkcji. Listy na przykład zawierające zdania w formie dialogu są krótkie — zawierają pytania i odpowiedzi z terminala. Realizacja procesów wysyłania i przyjmowania dużych programów wymaga współdziałania z systemami operacyjnymi dla obsługi wsadowej.



Rys. 2. Format listu (przykład)

Wzajemne związki i logikę procesów obsługi sieci przedstawia rys. 3. Całość zasad współpracy i realizacji programowej w sieci ujęta jest w zbiór reguł zwanych protokółami.

Własności protokółu zależą od konfiguracji sprzętu, podsieci komunikacyjnej i komputera użytkownika, a także od zadań i przeznaczenia sieci komputerowej. Wyrazem tego są np. różne protokoły przyjęte w sieciach ARPA, NPL i CYKLADES. Sieć ARPA łączy komputery użytkowników, wielodostępne, o rozbudowanej konfiguracji urządzeń abonenckich, służy interakcyjnym formom współpracy z siecią oraz zastosowaniom w działalności badawczej. Sieć NPL budowana jest w celu przesyłania dużych ilości danych. Sieć CYKLADES natomiast rzadziej łączy różnego typu komputery użytkowników, częściej zaś służy do obsługi zadań wsadowych i wykorzystywana jest do udostępniania bibliotek programów użytkowych.



Rys. 3. Hierarchia programów i protokoły współpracy w sieci komputerowej

WYMAGANIA UŻYTKOWE

Struktura i własności sieci komputerowej zależą przede wszystkim od jej przeznaczenia, mniej zaś od stosowanych środków.

Istota rozwiązań i dobór środków sprzętowo-programowych zależą od roli i celów, jakie ma spełnić sieć wobec abonentów. Ponieważ często występuje duża różnorodność potrzeb użytkowników, a sieci stanowią zbiór zróżnicowanych zasobów, wybór struktury sieci i stosowanych środków informatyki musi polegać na wzajemnym kompromisie.

W sieciach komputerowych specjalizowanych (orientowanych na określony zakres usług) problem ten — jak uprzednio zaznaczono — jest nieco prostszy. Dotychczasowe zastosowania rozwiązań sieciowych pozwalają wymienić pięć sposobów wykorzystywania sieci:

- zdalne uruchamianie zadań wiąże się z przenoszeniem gotowych do przetwarzania zadań do tych komputerów, w których znajdują się potrzebne zbiory danych
- zdalne przetwarzanie wsadowe wiąże się z przenoszeniem programów lub danych. Dane są zazwyczaj przesyłane do węzła, w którym znajduje się przetwarzany program, proces odwrotny natomiast występuje wówczas, gdy przetwarzany zbiór danych jest bardzo duży
- współpraca interakcyjna użytkowników z siecią z pulpitu urządzenia abonenckiego (terminala) umożliwia sterowanie jednym lub wieloma procesami przetwarzania wsadowego. Wyprowadzone interakcyjnie instrukcje lub żądania sterują przetwarzaniem i dostępem do zbiorów danych w celu realizacji w trybie konwersacyjnym krótkich zadań we wskazanym komputerze obliczeniowym sieci. Użytkownik może sterować z terminala pracą komputerów znajdujących się poza lokalnym węzłem sieci
- dynamiczny dostęp do zbiorów danych polega na tym, że przetworzone zadanie ma taki sam dostęp do odległych zbiorów danych jak do zbiorów lokalnych. Dzięki temu możliwa jest — w wypadku wykorzystania rozproszonej bazy danych — realizacja programu nie wymagająca od użytkownika specjalnego planowania, a nawet — teoretyczne stosowanie dowolnego typu interakcji między komputerami użytkowników sieci
- wzajemna wymiana obciążeń komputerów sieci wiąże się ze zrównoważeniem obciążeń. Polega ona na zdolności takiego przyjęcia i rozdzielenia danego zadania między komputery użytkowników sieci, aby można było uzyskać równomierne wykorzystanie zasobów sieci. Użytkownik, wprowadzając zadanie do sieci nie interesuje się, gdzie i jak zostanie ono zrealizowane. Całość zgłoszonych zadań użytkownika realizuje program sterujący siecią. Program ten znajduje wszystkie niezbędne zbiory danych, tworzy sekwencję sterującą zadaniem i przygotowuje zadanie do przetwarzania poprzez agregację zasobów sieci.

FUNKCJE I PARAMETRY SIECI

Omówione właściwości sieci i sposób ich użytkowania umożliwiają określenie ich funkcji i parametrów. Podstawową funkcją sieci komputerowej jest realizacja wymiany informacji między komputerami komunikacyjnymi, komputerami użytkowników i terminalami sieci w dowolnym czasie i w formie odpowiadającej wymaganiom użytkowników. Wyróżnić można funkcje związane z realizacją wymiany — zwane technicznymi i funkcje związane z obsługą użytkownika, zwane użytkowymi. Do pierwszych zalicza się funkcje:

— konwersji kodów
— konwersji szybkości i rodzaju transmisji
— sterowania przepływem informacji
— wykrywania i korekcji błędów.

Do drugiej grupy funkcji zalicza się następujące udogodnienia użytkowe:

— gromadzenie i aktualizację danych
— utajanie informacji
— dostarczanie zwłoczne
— wieloadresowanie, automatyczne i bezpośrednie adresowanie
— wielokrotne wywoływanie oraz zwrotne wywoływanie
— możliwość tworzenia w sieci wydzielonych, zamkniętych podsieci
— możliwość dostępu do informacji sieciowej
— możliwość współpracy z innymi sieciami.

Obecnie jeszcze nie wszystkie wymienione udogodnienia są dostępne.

Na ogół nie mogą one jeszcze być realizowane jednocześnie w tej samej sieci i dlatego prowadzi się dalsze badania dotyczące ich implementacji. Niemniej jednak można już stwierdzić, iż warunkami koniecznymi do zapewnienia właściwej realizacji tych funkcji sieci komputerowych są następujące parametry sieci:

— krótki czas reakcji
— duża szybkość i przepustowość transmisji
— niska stopa błędów
— przezroczystość kodowa (zgodność lub odpowiedzialność)
— wysoka niezawodność sieci jako całości
— niski koszt użytkowania sieci.

BADANIA I PROJEKTOWANIE SIECI

Największe doświadczenia oraz postęp w rozwiązaniach uzyskano w badaniu sieci realizujących funkcję obsługi dostępu do zasobów i wymiany programów. W użytkowaniu sieci przeważa tryb wsadowy i zapytaniowy. Sieci realizujące obsługę interakcyjną przeznacza się do zastosowań specjalnych, często nawet bardzo wąskich. Ponieważ tego rodzaju rozwiązania mają szczególne znaczenie dla doskonalenia obsługi użytkowników sieci, a ich zastosowania obejmują badania naukowe i dydaktykę, perspektywy ich rozwoju wiążą się z budową sieci uniwersyteckich i wojskowych. Jak zaznaczono na wstępie, kierunki tego rozwoju koncentrują się na doskonaleniu własności sieci komputerowych z punktu widzenia wymagań nauczania i badań oraz na zastosowaniach gospodarczych — w przemyśle i usługach. Sieci ARPA i CYKLADES, reprezentujące pierwszy kierunek zastosowań, stanowią bazę oryginalnych i cennych rozwiązań i dają jednocześnie podstawę do wypracowania naukowych metod budowy sieci komputerowych.

Dotychczasowy dorobek badawczy oraz zebrane doświadczenia w zakresie projektowania sieci komputerowych nie stwarzają dostatecznych podstaw metodycznych do budowy przyszłościowych struktur sieci komputerowych. Nadal wymagają rozwiązania:

— metody tworzenia rozproszonych baz danych w pamięciach różnych komputerów obliczeniowych sieci (zbiory danych o dużej pojemności)
— metody efektywnego dostępu do rozproszonych baz danych i ich dynamiczna modyfikacja (hierarchiczna organizacja gromadzenia danych i ich zabezpieczenie)

— metody sterowania przepływem zadań w sieci z punktu widzenia zdolności wykonywania określonego zadania w komputerze obliczeniowym sieci z wykorzystaniem takich zasobów, jak moc obliczeniowa, niezbędna wielkość pamięci, specjalistyczne oprogramowanie użytkowe (standaryzacja procedur sterujących)

— metody interakcyjnej współpracy użytkownika z siecią w wykorzystywaniu jej zasobów oraz rozwój metod sztucznej inteligencji (współpraca człowiek — maszyna).

*

Szybko postępująca komputeryzacja kraju, zwiększająca się liczba instalowanych wielodostępnych systemów komputerowych zarówno w szkolnictwie wyższym (WASC, CYFRONET) jak i w przemyśle oraz rozwój ilościowy i jakościowy zasobów stwarzają sprzyjające warunki, a zarazem konieczność budowy sieci komputerowych. Podjęto już prace w zakresie rozwiązań pilotowych sieci międzyuczelnianej Wrocław — Warszawa — Kraków oraz sieci na potrzeby administracji i przemysłu. Jednym z podstawowych czynników determinujących postęp w tej dziedzinie jest jakościowy rozwój krajowej bazy łączności telekomunikacyjnej, w której ostatnio uzyskano wiele liczących się w kraju osiągnięć.

Troska o przyszłość skłania do podjęcia prac mających na celu włączenie krajowej sieci komputerowej do sieci krajów socjalistycznych oraz innych krajów europejskich. Znaczenie tych przedsięwzięć jest ogromne, i to nie tylko ze względu na przewidywaną wymianę doświadczeń i osiągnięć w zakresie metod projektowania sieci, lecz przede wszystkim ze względu na wzajemne udostępnianie własnych zasobów sieci. Współpraca ta dotyczy koncepcji sieci projektowanych w ramach Międzynarodowej Komisji Współpracy Akademii Nauk Krajów Socjalistycznych w dziedzinie informatyki, Międzynarodowego Instytutu Analizy Systemów (IIASA) w Austrii zajmującego się budową wspólnej sieci dla krajów socjalistycznych i środkowo-europejskich oraz koncepcji dotyczących współpracy krajowej międzyuczelnianej sieci komputerowej z siecią CYKLADES.

Podjęte działania przyniosą ogromne korzyści w postaci intensyfikacji badań i tempa ich realizacji użytkowej oraz lepszą obsługę potrzeb użytkowników.

LITERATURA

[1] Computer Networks — The International Journal of Distributed Informatique, Volume 1, Number 1, June 1976

[2] IBM Systems Journal, Volume Fifteen, Number 1, 1976

[3] Internationales Seminar Komunikation in komplexen Informationsverarbeitungssystemen, Potsdam, Februar, 1976. (Komisja Współpracy Akademii Nauk Krajów Socjalistycznych w zakresie problemów naukowych informatyki — Grupa 03)

[4] Bazewicz M. — The utility functions of multiaccess timesharing systems and computer networks oraz materiały konferencji COMPUTER NETWORKS (IOK PAN i IIASA — Laxenburg), Zaborów, maj, 1976.

RYSZARD TADEUSIEWICZ
Akademia Górniczo-Hutnicza
Kraków

Język „MINI" jako propozycja w zakresie nauczania podstaw informatyki

Nauczanie informatyki zaczyna wchodzić jako przedmiot podstawowy do coraz liczniejszych programów studiów, kursów podyplomowych a także niektórych szkół średnich. Fakt ten uznać należy za bardzo pozytywny, równocześnie jednak zmusza on do poszukiwania nowych i efektywnych metod nauczania informatyki.

Dysponujemy już stosunkowo liczną kadrą, zdolną do prowadzenia zajęć dydaktycznych, istnieje względnie wyczerpujący podręcznik „propedeutyki informatyki", w większości szkół wyższych istnieje także możliwość udostępnienia studentom i słuchaczom kursów w ograniczonym zakresie czasu komputera. W tej sytuacji celowe jest skupienie uwagi na odpowiednim doborze treści przekazywanych w zakresie podstawowego kursu informatyki.

Wydaje się być rzeczą bezdyskusyjną, że jednym z ważniejszych zagadnień poruszanych w każdym podstawowym kursie podstaw informatyki jest nauka podstaw programowania. Dzieje się tak dlatego, że zdecydowana większość potencjalnych użytkowników sprzętu informatycznego widzi swoją ewentualną styczność z komputerem właśnie w charakterze programistów mniej lub bardziej złożonych zadań, a także dlatego, że dopiero samodzielna próba programowania nadaje całości wiedzy z zakresu podstaw informatyki konkretny i namacalny wyraz. Działają tu czynniki zarówno ściśle praktyczne (umiejętność samodzielnego programowania jest wszak coraz bardziej potrzebna), jak i (ważniejsze zapewne) czynniki emocjonalne: sukces, jakim jest udana próba samodzielnego uruchomienia programu i płynące stąd potwierdzenie teoretycznych wiadomości stwarzają bardzo korzystną motywację w całej dalszej nauce. Droga do pierwszego samodzielnego programu jest jednak na ogół długa, zbyt długa. Wieloletnie doświadczenia autora zebrane przy prowadzeniu licznych zajęć dydaktycznych z zakresu podstaw programowania pozwalają na wskazanie kilku powodów, dla których, początkujący student informatyki ma trudności z rozpoczęciem praktycznej działalności na maszynie.

Naukę programowania prowadzi się w oparciu o jeden z szeroko używanych języków algorytmicznych, takich jak FORTRAN, ALGOL 60, COBOL, BASIC. Są to języki o bardzo szerokich możliwościach, przystosowane do pisania skomplikowanych programów, w mniejszym jednak stopniu przystosowane do nauki podstawowej. Wynika to z kilku powodów:

- wymienione języki posiadają bardzo wiele różnorodnych instrukcji o często dość złożonej składni. Student przed napisaniem najprostszego nawet programu musi znaczną część tej złożoności opanować pamięciowo i koncepcyjnie
- złożoność, o której mowa, powoduje, że początkujący (student) staje przed koniecznością wyboru określonej formy spośród bardzo wielu ekwiwalentnych. Powoduje to zamieszanie i uczucie niepewności; w rezultacie student zazwyczaj zamiast myśleć poszukuje „wzorca", na którym mógłby się oprzeć
- duże trudności sprawiają w wymienionych językach operacje wejścia/wyjścia, których mnogość i złożona postać (FORMATY w języku FORTRAN!) pochłania nieproporcjonalnie wiele wysiłku

- wskazane języki implementowane są z konkretnymi translatorami, a translatory te wymagają określonej liczby instrukcji sterujących translacją (kart jobowych), które będąc z punktu widzenia merytorycznego (dla programisty) nieistotne, poważnie obciążają pamięciowo piszącego program. (Mowa naturalnie cały czas o programach szkoleniowych, bardzo małych i prostych)

- nie bez znaczenia jest też fakt, że słowa kluczowe (strukturalne) wymienionych wyżej języków są słowami języka angielskiego. Dla bardzo wielu studentów nie znających tego języka są to puste zestawy symboli, których znaczenia, a także ortografii trzeba się nauczyć na pamięć. Fakt ten sprawia trudności większej liczbie początkujących, niż się to powszechnie sądzi.

Przy pisaniu programów początkujący studenci popełniają błędy. Błędy te są wykrywane zarówno na etapie translacji, jak i na etapie wykonywania programu. W rezultacie student otrzymuje wydruk, zawierający standardową diagnostykę błędu, najzupełniej wystarczającą dla doświadczonego programisty i na ogół całkowicie nieczytelną dla stawiającego pierwsze kroki.

Trzeba więc wydruki diagnostyczne pracowicie rozkodowywać w oparciu o zamieszczane w literaturze wykazy błędów, przy czym na ogół okazuje się, że nawet po odczytaniu, jaki rodzaj błędu wykryto, wcale nie łatwo jest błąd zlokalizować. Pozostają więc żmudne konsultacje, poszukiwania — kształcące, ale i zniechęcające.

Podczas wykonywania na maszynie programów studenckich trzeba roztaczać nad nimi nadzór a ponadto trzeba prowadzić ewidencję wykonywanych programów i notatki odnośnie rodzaju niepowodzeń w przypadku ich przerywania w celu konsultowania i oceniania studentów. W praktyce prowadzi to przy licznych grupach i (jedynej słusznej) zasadzie „jeden student — jeden program" do bardzo dużej liczby prac o charakterze technicznym i administracyjnym.

Obciąża to personel prowadzący zajęcia dydaktyczne i w efekcie rzutuje na zmniejszenie efektywności nauczania. Przytoczone wyżej uwagi skłaniają do poszukiwania nowych dróg w nauczaniu podstaw programowania, pozwalających na ograniczenie niekorzystnego wpływu wymienionych czynników. Jednym z możliwych rozwiązań jest tu system, którego skrócony opis podano w niniejszym artykule. Podstawowymi założeniami tego systemu są następujące tezy:

- system powinien funkcjonować w oparciu o możliwie szeroko dostępny sprzęt informatyczny. Wybrano system ODRA 1300, jednak ze względu na to, że całe oprogramowanie systemu (translator, system operacyjny) zrealizowano w FORTRANIE jest możliwość łatwego przeniesienia także i na inne maszyny

- system powinien pracować w reżimie pracy wsadowej, gdyż tryb pracy konwersacyjnej przy istniejącej bazie sprzętu i rozmiarach istniejących potrzeb jest nieosiągalny

- system powinien charakteryzować się maksymalną prostotą, przy ograniczeniu do minimum liczby alternatywnych możliwości i dyrektyw administracyjnych obciążających programistę

- system przy swojej prostocie musi być maksymalnie zbliżony do powszechnie używanych języków algorytmicznych, tak aby nawyki uzyskane przy opanowywaniu podstaw programowania były później pomocne w rozwiązywaniu innych problemów

- system powinien umożliwiać pisanie względnie złożonych programów z wykorzystaniem typowych mechanizmów (pętle, użycie tablic, mechanizmy iteracyjne)

- słowa strukturalne powinny być słowami języka polskiego

- operacje wejścia/wyjścia powinny być maksymalnie uproszczone przy zagwarantowaniu czytelnych wydruków z programów

- system musi mieć rozległą i wyczerpującą diagnostykę błędów zarówno wykonania, jak i translacji, przy czym wykryty błąd musi być wyczerpująco skomentowany wydrukiem w języku polskim, wyjaśniającym rodzaj błędu, przyczyny jego powstania i metody jego poprawienia

- wykonywanie grupy programów studenckich powinno być w pełni zautomatyzowane, niezależnie od tego, ile programów liczy paczka i jakie ewentualne błędy (kompilacji, wykonania, niekompletne programy lub dane) znajdują się w programach

- system powinien nadzorować programy w czasie realizacji w celu wykrywania „zapętlonych" programów, które muszą być przerywane z odpowiednim komentarzem

- niezależnie od liczby programów w grupie, po zakończeniu pracy system powinien drukować raport dla prowadzącego zajęcia.

W raporcie należy umieścić informacje o nazwach programów, nazwiskach studentów, którzy je pisali, wykrytych błędach kompilacji lub wykonania (liczba i typy błędów), czasie wykonywania programów.

System, który niniejszym prezentuję spełnia wymienione wyżej postulaty. Składa się on z kompilatora i systemu operacyjnego, przy czym obydwa są programami napisanymi w języku FORTRAN.

System ten powstał w wyniku prac dyplomowych studentów: Haliny Kurpik, Barbary Czyczyło, Bartłomieja Szczurka i Andrzeja Drejaka. Język MINI, stanowiący trzon systemu posiada następujące właściwości:

Istnieje jedynie 9 typów instrukcji

PROGRAM	— instrukcja początkowa, deklarująca nazwę programu i nazwisko programisty
CZYTAJ	— instrukcja wprowadzania danych
DRUKUJ	— instrukcja wydruku wyników
SKOCZ DO	— instrukcja skoku bezwarunkowego
JEŚLI	— instrukcja skoku warunkowego
WYKONAJ	— instrukcja pętli
KONTYNUUJ	— instrukcja pusta
:=	— instrukcja podstawienia arytmetycznego
KONIEC	— instrukcja końca programu.

Wszystkie instrukcje są pisane według ustalonego formatu na kartach perforowanych, przy czym kolumny 1 i 2 karty służą do zapisu etykiet (etykiety wyłącznie cyfrowe), od 3 do 30 zapisuje się treść instrukcji, a kolumny od 32 do końca karty przeznaczone są na zapis komentarza.
Nazwy zmiennych i tablic są wyłącznie jednoliterowe, przy czym ustalono, że zmienne I, J, K, L, M, N są typu całkowitego, a pozostałe z wyjątkiem T i W oznaczają zmienne typu rzeczywistego. Nazwy T i W są zarezerwowane dla tablic, przy czym rozmiary tablic są przyjęte na stałe i wynoszą dla każdej z nich 99 elementów. (Obie dostępne w MINI tablice są jednowymiarowe). Odwołanie do elementu tablicy następuje przez wypisanie nazwy tablicy wraz z indeksem, przykładowo T3, W16, WI, TK itp.

Jak widać nazwa tablicy nie jest tu oddzielona od wskaźnika żadnym znakiem (nawiasem). Przy pisaniu instrukcji w języku MINI można natomiast dowolnie redagować zapis za pomocą spacji, które podobnie jak w języku FORTRAN nie mają żadnego znaczenia.

Omówimy teraz semantykę instrukcji języka MINI, opis formalny w zapisie Backusa znaleźć można w pracy [2]. Instrukcja PROGRAM deklaruje nazwę programu i nazwisko programisty (studenta). Postać instrukcji:

PROGRAM nazwa nazwisko

przy czym **nazwa** oznacza nazwę programu (maks. 10 znaków), a **nazwisko** (wpisywane jako komentarz od 32 kolumny) — 20-znakowy zapis nazwiska programisty.
Instrukcje wejścia/wyjścia mają postać:

- CZYTAJ lista nazw zmiennych jednakowego typu

- DRUKUJ lista nazw zmiennych.

Przy czytaniu obowiązują takie zasady, jak przy czytaniu formatem swobodnym, to znaczy rozmieszczenie liczb na zewnętrznym nośniku informacji jest obojętne, byle tylko każda liczba było oddzielona od pozostałych przynajmniej jedną spacją. Z opisu instrukcji CZYTAJ wynika, że nie wolno czytać jedną instrukcją równocześnie liczb całkowitych i rzeczywistych. Ograniczenie to nie obowiązuje przy drukowaniu. Przy wykonywaniu instrukcji DRUKUJ dla każdej pozycji listy wydrukowany zostaje na drukarce wierszowej jeden wiersz postaci

$$V = w$$

gdzie V jest nazwą zmiennej a w jej wartością. W skład listy w instrukcjach wejścia/wyjścia wchodzić mogą ele-

menty tablic. Instrukcja SKOCZ DO jest instrukcją skoku bezwarunkowego o postaci:

SKOCZ DO etykieta

gdzie **etykieta** musi znajdować się przy jednej z instrukcji tego samego segmentu, w którym wystąpiła instrukcja SKOCZ DO. Instrukcja skoku warunkowego JEŚLI ma postać:

JEŚLI (warunek) SKOCZ DO etykieta

Warunek ma tu postać relacji równości (znak =), większości (znak >) lub mniejszości (znak <) pomiędzy dwoma wyrażeniami arytmetycznymi. Postacie dopuszczalnych w MINI wyrażeń arytmetycznych omówione będą w punkcie poświęconym instrukcji podstawienia.

Instrukcja pętli ma postać następującą:

WYKONAJ etykieta $i := m_1$ DO m_2 KROK m_3

gdzie **etykieta** jest etykietą ostatniej instrukcji objętej pętlą, i oznacza całkowitą zmienną kontrolną pętli, a parametry m_1, m_2, m_3 oznaczają odpowiednio wartość początkową, końcową i krok zmian zmiennej kontrolnej. Jeśli krok m_3 wynosi 1 to można go pominąć wraz ze słowem strukturalnym KROK. Fakt taki jest komentowany podczas translacji wydrukiem, w którym translator informuje, że przyjęto krok równy 1. Przy posługiwaniu się pętlą WYKONAJ obowiązują ogólnie przyjęte ograniczenia odnośnie skoków do wnętrza pętli itd. Wartości parametrów m_1, m_2, m_3 muszą być dodatnie, a same parametry muszą być stałymi lub zmiennymi typu całkowitego.

Instrukcja KONTYNUUJ jest instrukcją pustą; jako instrukcja adresowalna może być zaopatrzona w etykietę i stanowić np. ostatnią instrukcję pętli WYKONAJ (która nie może się formalnie kończyć na żadnej instrukcji skokowej).

Instrukcja podstawienia ma postać ogólną:
$V := w$

gdzie V jest nazwą zmiennej lub elementem tablicy, a w jest wyrażeniem arytmetycznym. W języku MINI wyrażenia arytmetyczne mogą być dowolnie złożone i mogą zawierać zarówno elementy typu rzeczywistego, jak i typu całkowitego. Typ wyrażenia jako całości jest ustalany jako całkowity jeśli **wszystkie** elementy tego wyrażenia są typu całkowitego, inaczej typ jest rzeczywisty.

Ma to tę ważną konsekwencję, że w wyrażeniach typu całkowitego wszystkie operacje częściowe są wykonywane jako całkowite (z odrzuceniem części ułamkowej wyników).

W języku MINI obowiązuje przy wykonywaniu wyrażeń arytmetycznych kolejność operacji:

— obliczanie wartości funkcji i wyrażeń w nawiasach
— potęgowanie (oznaczane operatorem ↑)
— mnożenie i dzielenie (oznaczane odpowiednio * i /)
— dodawanie i odejmowanie (oznaczane + i —).

Wewnątrz każdej z wymienionych klas obowiązuje porządek obliczania od lewej do prawej. W języku MINI dostępnych jest 12 funkcji jednoargumentowych typu rzeczywistego: SIN, COS, TAN (tangens), ASI (arcus sinus), ACO (arcus cosinus), ATN (arcus tangens), ACT (arcus cotangens), ABS (wartość bezwzględna), PWK (pierwiastek kwadratowy, odpowiednik powszechnie używanej, a bardzo „egzotycznej" dla początkujących funkcji SQRT), EXP, LGN (logarytm naturalny), LGD (logarytm dziesiętny). Wywołanie dowolnej funkcji następuje w wyniku użycia w wyrażeniu arytmetycznym nazwy funkcji z argumentem będącym stałą, zmienną lub elementem tablicy. Argument funkcji umieszczany jest w nawiasach. Na uwagę zasługuje fakt, że przy całej prostocie struktury języka MINI zachowano możność używania dowolnie złożonych wyrażeń arytmetycznych, z użyciem nawiasów i funkcji. Wynikało to z obserwacji, że przy nauczaniu studentów programowania w językach zawierających ostre restrykcje na postać dopuszczalnych wyrażeń (np. autokod MOST na EMC ODRA 1013), musieli oni wiele uwagi poświęcić rozbijaniu bardziej złożonych wzorów na proste operacje, wyrabiając sobie złe nawyki i pisząc programy dalekie od optymalności.

Instrukcja KONIEC oznacza koniec programu (informacja dla kompilatora), a równocześnie zatrzymuje pracę maszyny nad danym programem w fazie wykonywania.

Podane wyżej zasady pisania programów najlepiej zilustruje przykład konkretnego programu napisanego w języku MINI:

```
PROGRAM SIMPSON                    CZERSKI ADAM
CZYTAJ A,B                         PRZYKLADOWY PROGRAM
CZYTAJ N                           W JEZYKU MINI
H:= (B — A) /N                     OBLICZA CALKE
                                   Z FUNKCJI
C:= COS (A) + COS (B) / B          COS (X) / X
N:= N — 1                          W ZADANYCH GRANICACH
                                   [A, B]
K:= 4                              METODA SIMPSONA
WYKONAJ 15 I:=1 DO N KROK 1
A:= A + H
C:= C + K * COS (A) / A
JESLI (K = 4) SKOCZ DO 12
K:= 4
SKOCZ DO 15
12K:= 2
15KONTYNUUJ
C:= C * H/3                        C — WARTOSC CALKI
DRUKUJ C
KONIEC
```

Jak wspomniano wyżej, system obok kompilatora (zrealizowanego w technice mieszanej — interpretacyjno-kompilacyjnej) zawiera minisystem operacyjny obsługujący translację i wykonanie większej partii zadań studenckich a po zakończeniu wykonania dokonujący obszernej sprawozdawczości, będącej kompletną dokumentacją dla prowadzącego zajęcia. W systemie tym przewidziano rozmaite błędy powstałe przy przygotowywaniu paczki programów, a więc obok błędów formalnych (wykrywanych przez translator) i merytorycznych (wykrywanych w trakcie wykonywania) uwzględniono także możliwość umieszczenia w paczce programu niekompletnego, programu z za małą lub zbyt dużą liczbą danych itp.

Wszystkie wymienione przypadki są przez system wykrywane, komentowane i nie mają wpływu na wykonywanie innych programów tej samej paczki. System przewiduje też przypadek kiedy liczba programów paczki jest zbyt duża. W takim przypadku na konsolę operatora wyprowadzane są dyspozycje, których wykonanie powoduje rozdzielenie paczki i wykonanie jej w dwu seriach.
Jedną z najistotniejszych zalet systemu jest obszerna informacja w języku polskim dołączana do każdego błędnego programu.

Translator języka MINI wykrywa 59 rodzajów błędów, z których każdy posiada swój typowy komentarz. Błędy są zliczane i liczba wykrytych błędów podawana jest na końcu listingu błędnego programu jak również w raporcie dla prowadzącego zajęcia. Poniżej podano przykładowo postać trzech wydruków podawanych po wykryciu błędu kompilacji (przytoczony fragment jest przedrukiem z rzeczywistego listingu kompilacji: podano też instrukcje, które wywołały komentarze)

```
PROGRAM

** 1 BLAD — BRAK NAZWY PROGRAMU

PO WYRAZIE „PROGRAM" MUSI WYSTAPIC NAZWA PROGRAMU W POSTACI CIAGU ZNAKOW ROZNYCH OD BLANKU UMIESZCZONYCH PRZED KOLUMNA 31.

1 SKOCZ DO 17

WYKONAJ 1 N:= 1 DO 4

** 2 BLAD — ETYKIETA KONCA PETLI WYSTAPILA JUZ WCZESNIEJ

17? := 3

** 3 BLAD — NIEROZPOZNANY POCZATKOWY ZNAK INSTRUKCJI. W INSTRUKCJI ARYTMETYCZNEJ WARTOSC WYRAZENIA MOZNA PODSTAWIAC POD ZMIENNA PROSTA A —Z LUB ELEMENT TABLICY T LUB W.

KONIEC

** 4 BLAD — BRAK ETYKIETY 1

** 5 BLAD — SKOK DO ETYKIETY 17 Z ZEWNATRZ PETLI. WEJSCIE DO ZAKRESU PETLI Z ZEWNATRZ NIE JEST DOZWOLONE.

***** KONIEC KOMPILACJI — 5 NIEPOPRAWNYCH INSTRUKCJI.
```

Jak widać z przytoczonego przykładu, w większości przypadków komentarze maszyny na tyle dokładnie wyjaśniają sytuację, że student jest w stanie samodzielnie poprawić program. Podobnie sytuacja wygląda, gdy w czasie wykonywania programu wystąpi błąd wykonania. Program jest

wówczas przerywany i wyprowadzana jest szczegółowa informacja o błędzie, jego rodzaju, lokalizacji i prawdopodobnych przyczynach. Wyprowadzany jest też „ślad" błędnego programu (odpowiednik „TRACE" z oprogramowania ODRA 1300, ale w języku polskim i z komentarzami). Zachowanie maszyny w przypadku wykrycia błędu wykonania zilustruje podany niżej konkretny przykład:

```
PROGRAM MASZT                    JAWORZAK KRZYSZTOF
CZYTAJ I
DRUKUJ I
K:= I ↑ 10
DRUKUJ K
KONIEC
***** KONIEC KOMPILACJI — PROGRAM NAPISANY POPRAWNIE. *****
OTRZYMANE WYNIKI
* * * * * * * * * * * * * * * * * * *
I = 125
BLAD WYKONANIA
* * * * * * * * * * * * * * * * * * *
```

W CZASIE WYKONYWANIA OPERACJI ARYTMETYCZNEJ WYSTAPIL NADMIAR. NADMIAR MOZE WYSTAPIC, GDY WYNIK WYRAZENIA ARYTMETYCZNEGO JEST POZA DOPUSZCZALNYM ZAKRESEM LUB GDY ROZNICA MIEDZY WARTOSCIAMI WYRAZEN ARYTMETYCZNYCH POROWNYWANYCH W INSTRUKCJI „JESLI" PRZEKRACZA TEN ZAKRES. ZAKRES TEN TO DLA LICZB CALKOWITYCH PRZEDZIAL: [—8388607, + 8388607], A DLA LICZB RZECZYWISTYCH PRZEDZIAL: [—5.6E76, + 5.6E76.]. NADMIAR WYSTAPI TEZ GDY WARTOSC RZECZYWISTA Z PRZEDZIALU DOPUSZCZALNEGO DLA LICZB RZECZYWISTYCH A PRZEKRACZAJACA ZAKRES LICZB CALKOWITYCH PODSTAWIA SIE POD ZMIENNA CALKOWITA.

INFORMACJA O OSTATNICH POPRAWNIE WYKONANYCH INSTRUKCJACH:

—2 CC 1

—1 DC 1

JEDEN WIERSZ DOTYCZY JEDNEJ INSTRUKCJI I ZAWIERA NASTEPUJACE INFORMACJE (ODPOWIEDNIO W KOLUMNACH):

1 — NUMER WIERSZA W ZAKRESIE OD —25 DO —1

2 — ETYKIETE INSTRUKCJI (JEZELI JEST) LUB SPACJE

3 — SKROT NAZWY INSTRUKCJI

4 — WARTOSC LICZBOWA LUB LOGICZNA INSTRUKCJI ALBO SPACJE, JESLI INSTRUKCJA NIE MA WARTOSCI.

SKROTY I WARTOSCI INSTRUKCJI:

INSTRUKCJA	— SKRÓT —	WARTOŚĆ
PODSTAWIENIE ARYTMETYCZNE	— AR —	WARTOSC LEWEJ STRONY PODSTAWIENIA
CZYTAJ (LISTA CALKOWITA)	— CC —	ILOSC ELEMENTOW LISTY WEJSCIA
CZYTAJ (LISTA RZECZYWISTA)	— CR —	ILOSC ELEMENTOW LISTY WEJSCIA
DRUKUJ (LISTA CALKOWITA)	— DC —	ILOSC ELEMENTOW LISTY WYJSCIA
DRUKUJ (LISTA RZECZYWISTA)	— DR —	ILOSC ELEMENTOW LISTY WYJSCIA
SKOCZ DO	— SK —	
JESLI	— JE —	TAK — GDY WARUNEK SPELNIONY
		NIE — GDY WARUNEK NIE SPELNIONY
WYKONAJ	— WY —	WARTOSC ZMIENNEJ KONTROLNEJ
KONTYNUUJ	— KO —	

Jak widać dołożono starań, aby diagnostykę błędu uczynić maksymalnie zrozumiałą i czytelną. Przytoczone przykłady mogą sprawiać wrażenie, że liczba komentarzy systemu jest przesadna („rozgadany"). Trzeba jednak mieć na względzie fakt, że przytoczone w charakterze przykładów programy były skrajnie proste, podczas gdy zwykle pisane przez studentów programy są kilkakrotnie dłuższe, zawierają mniejszą liczbę błędów, niż podany wyżej przykład i błędy te nie zawsze są tak banalne.

Dodatkowo należy pamiętać o dydaktycznym przeznaczeniu systemu, a także mieć na uwadze fakt, że dzięki odpowiedniemu selekcjonowaniu przytoczonych napisów w programie translatora i systemu operacyjnego, zajmują one względnie niewiele miejsca w pamięci i pracują stosunkowo szybko (translacja i wykonanie paczki 30 programów studenckich o typowym stopniu trudności trwa na EMC ODRA 1304 około pół godziny). Czas pracy maszyny nad jednym programem jest limitowany i wynosi 5 minut. Po upływie tego czasu program jest przerywany z odpowiednim komentarzem i automatycznym wyprowadzeniem „śladu", po czym maszyna przystępuje do obliczania kolejnego programu. Typowy program studencki liczy około 100 sekund.
Przedstawiona w artykule metoda nauczania programowania w ramach nauczania podstaw informatyki nadaje się wyłącznie dla kursów o ograniczonym czasie trwania i małej szczegółowości.

W przypadku dysponowania dłuższym czasem wskazane jest jednak także zapoznanie studentów z prostą metodą programowania, jaką oferuje system MINI przed rozpoczęciem nauki bardziej złożonych metod programowania, gdyż można dzięki temu uzyskać szybkie przełamanie „bariery psychologicznej" utrudniającej zwykle pierwsze kroki w nauce programowania. Oczywiście w tym drugim przypadku MINI stanowi tylko etap przejściowy do programowania w bardziej zaawansowanych systemach, takich jak FORTRAN czy ALGOL.

Wydaje się, że prostota języka i systemu MINI predestynuje go także do nauczania w ramach kursów informatyki w szkołach średnich, a nawet (w ograniczonym zakresie) ewentualnie w szkołach podstawowych.

Wydaje się także, że przystosowanie języka MINI do najpopularniejszego w Polsce systemu ODRA 1300 powinno ułatwić dostęp do praktycznej realizacji programów w tym języku, a wysoka efektywność translatora zapewnia minimalne koszty nauczania prowadzonego tą metodą. Autor niniejszego opracowania oferuje wszystkim zainteresowanym translator MINI w postaci binarnej w zamian za uwagi o jego pracy.

LITERATURA

[1] Kurpik H., Szczurek B. — Koncepcja prostego języka programowania EMC do celów dydaktycznych. Praca magisterska, AGH, Kraków 1975

[2] Czyczyło B., Drejak A. — Translator prostego języka algorytmicznego. Praca magisterska, AGH, Kraków 1976

[3] Gries D. — Compiler Construction for Digital Computers. Mc Graw Hill, New York 1974

[4] Tadeusiewicz R. — Założenia nauczania podstaw informatyki. Opracowanie wewnętrzne Instytutu Informatyki i Automatyki AGH, Kraków 1975.

Krajowa konferencja maszyn matematycznych w USA

W dniu 13—16 czerwca 1977 r. w Dallas w stanie Teksas odbędzie się konferencja poświęcona zagadnieniom przetwarzania danych. Jednocześnie czynna będzie wystawa sprzętu, na której zaprezentowane zostaną najnowsze osiągnięcia techniki, a w tym większe jednostki centralne, minikomputery, urządzenia peryferyjne, oprogramowanie, mikroprocesory, pamięci, terminale, usługi itp. Wystawa połączona będzie z cyklem seminariów technicznych. Planuje się szereg sesji z udziałem producentów komputerów, użytkowników i kierowników przedsiębiorstw.

(AEK)

KRZYSZTOF MARCINIAK
BARBARA PUTZ
Instytut Automatyki Przemysłowej
Politechniki Warszawskiej

Realizacja systemu APO-3

System programowania frezarek sterowanych numerycznie APO-3 został zaprojektowany i zrealizowany w Instytucie Automatyki Przemysłowej Politechniki Warszawskiej na zlecenie Centrum Badawczo-Konstrukcyjnego Obrabiarek w Pruszkowie.

Zadaniem systemu jest skrócenie czasu programowania, wyeliminowanie nużących i pracochłonnych czynności związanych z procesem przygotowania dokumentacji technologicznej dla frezarek sterowanych numerycznie oraz podniesienie niezawodności i wydajności tego procesu.

Przyjęto, że dane dla systemu stanowi program źródłowy opisujący obróbkę części. Program ten sformułowany jest w języku APO-3. Język ten można zakwalifikować do grupy języków APT-podobnych. System przeznaczony jest do generowania oprogramowania technologicznego dla grupy frezarek o podobnej kinematyce obróbki. Z tego względu przetwarzanie programu źródłowego przebiega w dwu etapach:

— w pierwszym wykonane zostają obliczenia opisane w programie źródłowym i związane ze wspólnymi cechami wszystkich obrabiarek (etap ten wykonuje fragment systemu nazywany procesorem)

— w drugim wyniki z procesora dostosowane zostają do wymagań konkretnej obrabiarki za pomocą specjalnego programu nazwanego postprocesorem tej obrabiarki.
Schemat przetwarzania przedstawiony jest na rys. 1.

ORGANIZACJA SYSTEMU

Procesor składa się z translatora i interpretera. Translator analizuje program źródłowy, sprawdza jego poprawność formalną oraz generuje program w postaci pośredniej (w języku pośrednim) przeznaczonej do interpretacji.

Budowa języka pośredniego zbliżona do zapisu wyrażeń arytmetycznych w odwrotnej notacji polskiej związana jest ściśle ze strukturą interpretera. Interpreter może być opisany jako maszyna wirtualna ze zbiorem instrukcji odpowiadających operacjom języka pośredniego i realizacją programu za pomocą techniki stosowej.

Ze względu na czynniki nie dające się przewidzieć w trakcie projektowania i realizacji przyjęto koncepcję systemu otwartego. Oznacza to, że struktura języka zbliżona do uproszczonej wersji FORTRANU pozostaje niezmienna, natomiast operacje języka interpretowane są przez niezależne i wymienne podprogramy. Zarówno translator i interpreter składają się z części sterującej oraz zbioru podprogramów translujących (lub interpretujących) odpowiednie sformułowania językowe. Modyfikacja języka źródłowego (na przykład dodanie nowej czynności) pociąga za sobą dodanie do translatora podprogramu przetwarzającego odpowiednie sformułowanie językowe, dodanie podprogramu interpretującego do interpretera oraz drobne zmiany w tablicach sterujących interpretacją i translacją.

ORGANIZACJA ZESPOŁU REALIZUJĄCEGO

Trzyetapowa struktura przetwarzania (translacja, interpretacja, postprocesorowanie) oraz fakt jednorazowego przekazywania danych między etapami pozwala realizować niemal niezależnie trzy podstawowe bloki strukturalne systemu, którymi są translator, inrpreter i postprocesory.

Oprócz podziału systemu na trzy bloki narzuca się podział bloków translacji i interpretacji według prostej struktury hierarchicznej. Podstawowym elementem bloku jest program, który steruje działaniem podrzędnych, wzajemnie niezależnych i nie komunikujących się ze sobą podprogramów.

Przyjęta struktura translatora i interpretera umożliwia realizację podprogramu na podstawie jego opisu funkcjonalnego oraz założeń co do postaci danych wymienianych między podprogramem a odpowiednim programem sterującym.

Realizację systemu podzielono na pięć zadań:
- opracowanie programu sterującego translacją
- opracowanie podprogramów translacji
- opracowanie programu sterującego interpretacją
- opracowanie podprogramów interpretacji
- opracowanie postprocesorów.

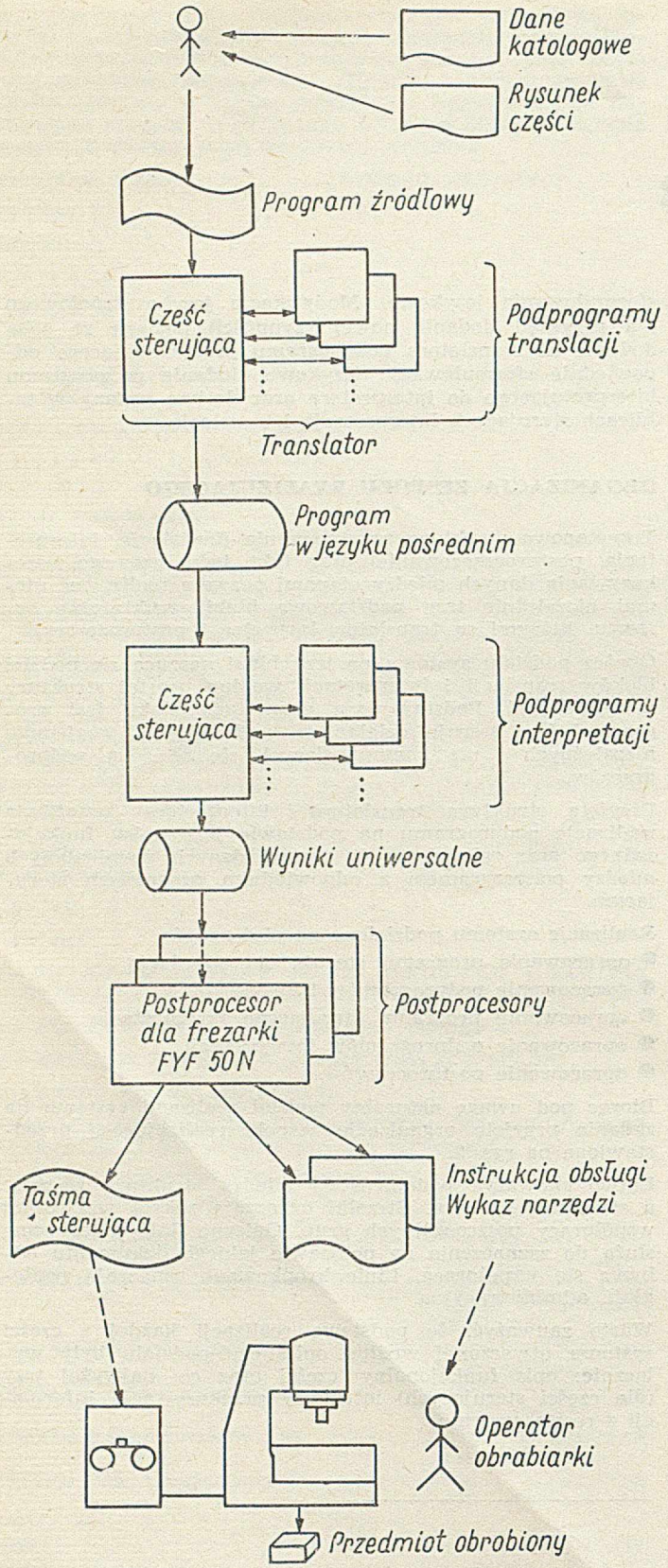
Biorąc pod uwagę naturalny podział realizacji systemu na zadania przyjęto organizację zespołu realizującego przedstawioną na rys. 2.

Łatwo zauważyć podobieństwo między strukturą zespołu, a strukturą systemu. Strzałki oznaczają zakres wzajemnej współpracy poszczególnych grup. Opisane linie przerywane służą do zaznaczenia na podstawie jakiego dokumentu odbywa się współpraca. Linie kropkowane oznaczają podległość administracyjną.

Warto zauważyć, że podstawą realizacji każdej z części systemu utworzonej według opisanego podziału były wyłącznie: opis funkcjonalny części oraz co najwyżej trzy (dla części sterujących) interfejsy przekazywania informacji z resztą systemu.

Dr inż. KRZYSZTOF MARCINIAK ukończył studia w roku 1969 na wydziale Mechaniki Precyzyjnej Politechniki Warszawskiej w zakresie automatyki. W roku 1973 ukończył studia na wydziale Matematyki i Fizyki Uniwersytetu Warszawskiego w zakresie matematyki. W roku 1975 uzyskał stopień doktorski za pracę dotyczącą automatyzacji programowania obrabiarek sterowanych numerycznie. Pracuje w Instytucie Automatyki Przemysłowej Politechniki Warszawskiej. Zajmuje się zagadnieniami oprogramowania urządzeń sterowanych numerycznie.

Mgr inż. BARBARA PUTZ ukończyła wydział Mechanizacji Precyzyjnej Politechniki Warszawskiej w roku 1973. Od tego czasu pracuje w Instytucie Automatyki Przemysłowej PW. Kierowała realizacją interpretera w systemie APO-3. Obecnie zajmuje się metodami komputerowego zapisu i projektowania powierzchni technicznych.



Rys. 1. Schemat przetwarzania programu w systemie APO-3

PRZEBIEG REALIZACJI

Liniowa, ogólnie rzecz biorąc, struktura przetwarzania za pomocą kolejnych bloków funkcjonalnych sugerowała uruchamianie interpretacji po translacji, a postprocesorów po interpretacji.
Przedstawiona strategia była nie do przyjęcia co najmniej z dwu powodów:
— oczekiwanie bloków na swoją kolej uruchomienia spowodowałoby zbyt długi okres realizacji
— nierównomierne obciążenie realizatorów poszczególnych części w czasie powodowałoby trudności organizacyjne (np. trudności we wprowadzaniu zmian) oraz wzrost kosztów.

Postanowiono uruchamiać wszystkie trzy bloki funkcjonalne równocześnie.

Opisy funkcjonalne oraz zasady przekazywania informacji między częściami systemu opracowano na etapie projektowania. Pozostał problem generowania danych testowych dla kolejnych części systemu. Realizacja szeregowa wymagałaby uruchomienia i wytestowania wszystkich bloków poprzednich.

W przypadku translacji dane stanowiły programy źródłowe, których przygotowanie niejako z definicji nie sprawiało trudności.

W przypadku interpretacji dane musiały być wygenerowane w pamięci masowej tworząc program w postaci pośredniej. Aby przyspieszyć uruchamianie interpretacji zrealizowano generator danych KLAP. KLAP był programem symulującym translację, który na podstawie danych liczbowoznakowych dostarczonych przez programistę, generował stan pamięci komputera odpowiadający przetłumaczeniu pewnego programu z języka źródłowego na pośredni.

Problem danych testowych dla postprocesorów nie nastręczał podobnych kłopotów. Miały one postać ciągu liczb i czytelną intuicyjnie strukturę, tak że przygotowanie ich pozostawiono bezpośrednio programistom. Rys. 3 przedstawia wykres czasowy realizacji systemu. Linią grubą zaznaczono drogę krytyczną, a cyfry nad strzałkami oznaczają w sposób zgodny z rys. 3 zaangażowanie poszczególnych grup realizatorów.

Montaż systemu odbywał się w następujących czterech etapach:

Etap 1

— uruchamianie programu sterującego translacją (testowanie za pomocą prostych programów w języku źródłowym)
— uruchamianie podprogramów translacji (testowanie przy użyciu danych sporządzanych indywidualnie)
— uruchamianie programu sterującego interpretacją (testowanie za pomocą generatora KLAP)
— uruchamianie podprogramów interpretacji (testowanie za pomocą danych sporządzanych indywidualnie, częściowo z użyciem generatora KLAP)

Etap 2

— montaż bloku translacji (dołączanie podprogramów do części sterującej)
— montaż bloku interpretacji
— uruchamianie postprocesorów

Etap 3

— uruchamianie, testowanie i modyfikacja procesora

Etap 4

— uruchamianie, testowanie i modyfikacje systemu (dodanie postprocesorów).

Problem wyszukiwania błędów na etapie uruchamiania i przypisywania ich odpowiedniemu programiście rozwiązano stosując trzypoziomowy system monitorowania.

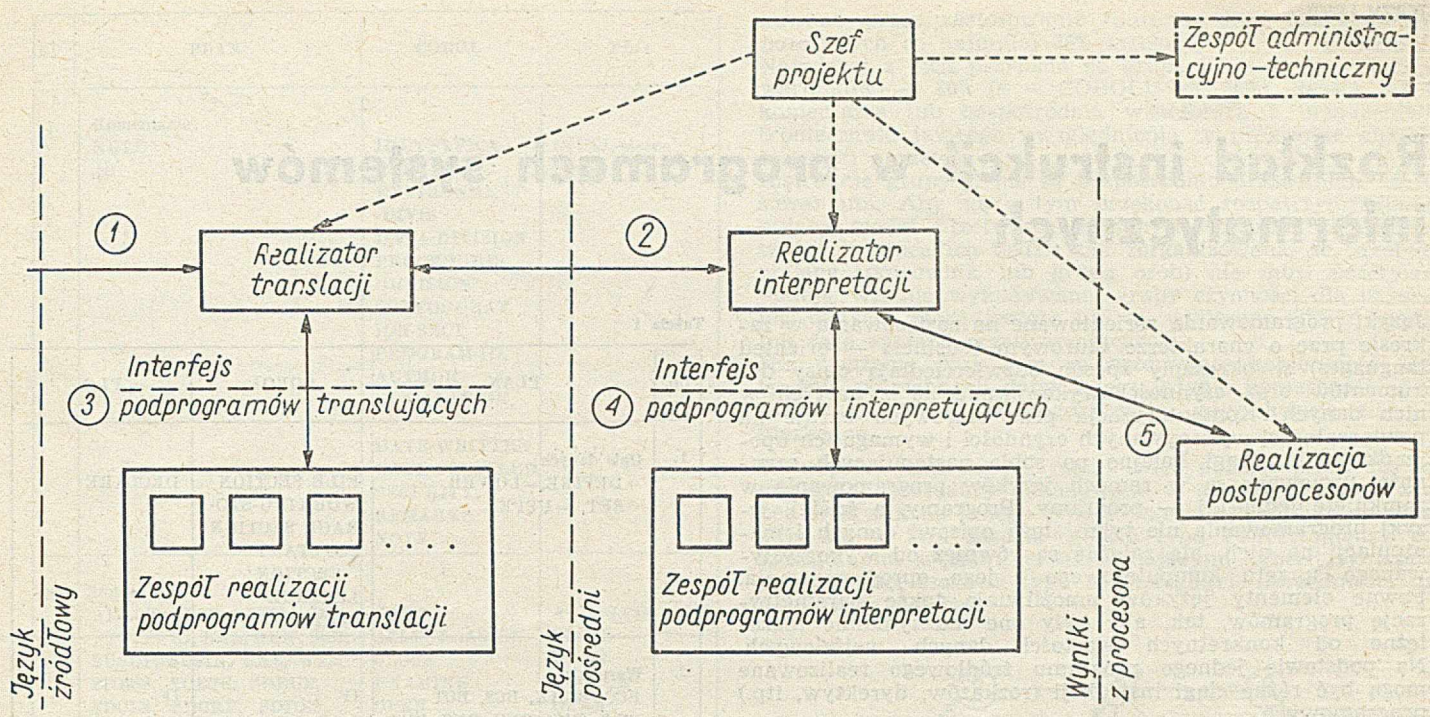
Poziom 1 — wydruki między modułami pozwalały określić, który z modułów powodował błąd

Poziom 2 — wydruki współpracy z podprogramami sporządzane przez program sterujący pozwalały zlokalizować błędny podprogram (lub program sterujący) oraz dostarczyć autorowi dokumentację błędu

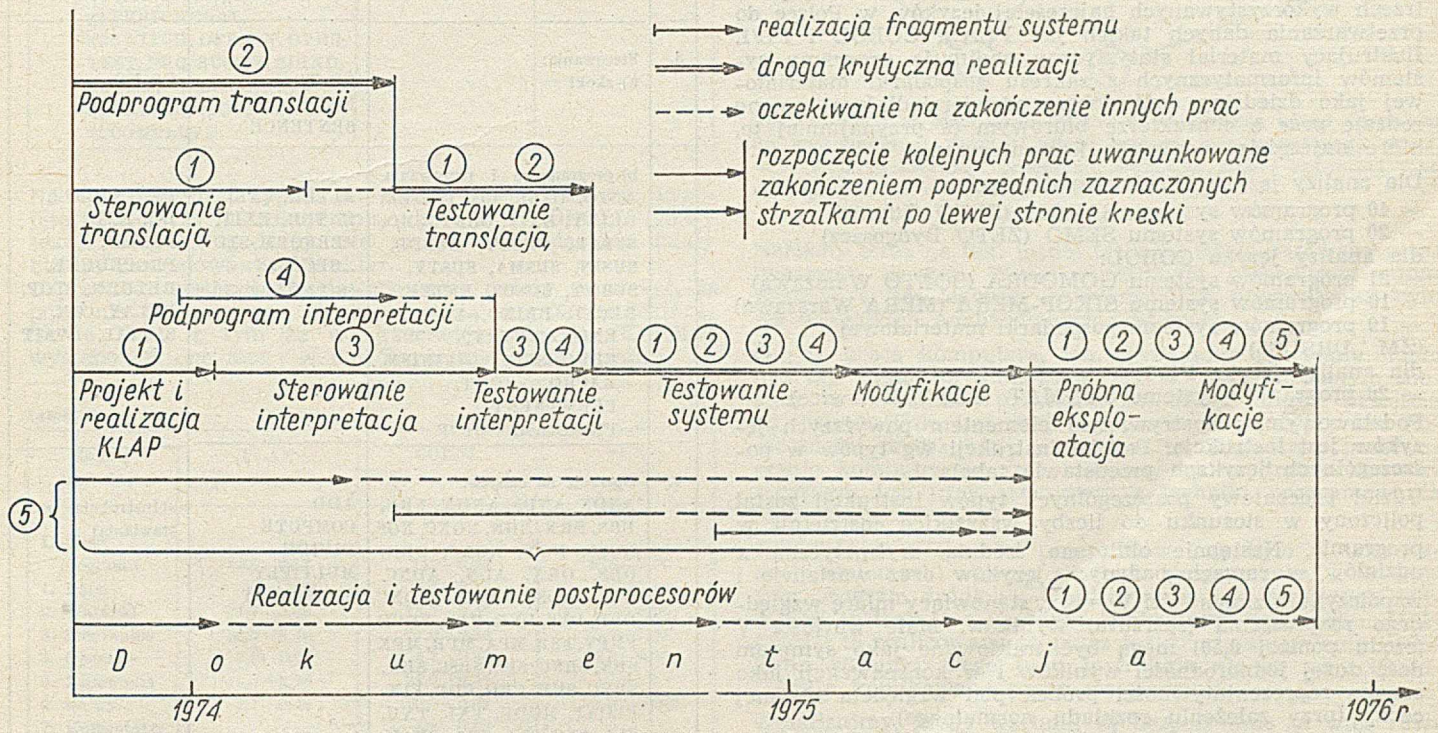
Poziom 3 — wydruki wewnętrzne generowane przez podprogramy i programy sterujące, pozwalające ich autorom śledzić szczegółowo przebieg przetwarzania.

Działanie przedstawionego mechanizmu wyszukiwania i eliminacji błędów okazało się w pełni zadowalające.

Przeprowadzenie modyfikacji w trakcie uruchamiania systemu powodowało powstawanie problemu jednoczesnych zmian, prowadzącego do konfliktów typu, złe działanie testu wynikające ze zmian wprowadzonych przez kogo innego a jeszcze nie naniesionych w dokumentacji itp. Aby temu zapobiec modyfikację systemu przeprowadzono generując kolejne jego wydania (obecnie działa wydanie 5).



Rys. 2. Realizacja zespołu realizującego



Rys. 3. Harmonogram realizacji systemu APO-3

Poprawki i zauważone błędy wpisywano do księgi poprawek opisując sposób ich poprawienia w następnym wydaniu. Testy systemu odnosiły się tylko do konkretnego wydania i przed przystosowaniem ich do następnego autor miał obowiązek wprowadzenia modyfikacji wynikającej z księgi poprawek. Przedstawiona procedura wprawdzie wprowadzała pewien porządek, ale wyraźnie wydłużała proces uruchamiania systemu. Ponieważ krótki czas realizacji systemu był jednym z podstawowych wymagań w okresach między poszczególnymi wydaniami dopuszczono również jednoczesne wprowadzanie kilku wzajemnie zależnych zmian na zasadzie działania w grupach nieformalnych. W kilku przypadkach jedynym wynikiem było zamieszanie, jednak w całości posunięcie to przyczyniło się do skrócenia cyklu uruchamiania.

Na zakończenie pragniemy zwrócić uwagę na konieczność zapewnienia stabilności zespołu realizującego. Na najbardziej niewdzięcznym etapie uruchamiania i modyfikacji wszelkie zmiany osobowe były niedopuszczalne.

LITERATURA

[1] Instrukcja programowania APO-3, Wydanie wewnętrzne CBKO, 1976 r.
[2] Automatyczne programowanie obrabiarek w języku APO-3, Mechanik, luty 1976 r.
[3] System automatyzacji programowania frezarek sterowanych numerycznie APO-3. Prace III Krajowej Konferencji Informatyków, tom 3, zeszyt 5, Katowice 1976 r.

JERZY LEYK
Warszawa

Rozkład instrukcji w programach systemów informatycznych

Języki programowania zorientowane na zastosowania w zakresie prac o charakterze biurowym (business — oriented languages) w określony sposób odzwierciedlają opisy dokumentów oraz czynności wykonywane na zawartych w nich danych. Komputeryzacja prac biurowych bazuje na powtarzalności wykonywanych czynności i wymaga ich uporządkowania. Ciągi kolejno po sobie następujących czynności ujmowane są w ramach języków programowania w zamknięte jednostki — programy. Programy, a ściślej, języki programowania nie tylko służą opisowi danych i manipulacji na nich, ale zależne są również od wykorzystywanego sprzętu komputerowego i jego oprogramowania. Pewne elementy języków umożliwiają także parametryzację programów, tak, aby były one maksymalnie niezależne od konkretnych wartości danych wejściowych. Na podstawie jednego programu źródłowego realizowane mogą być różne ciągi instrukcji (rozkazów, dyrektyw, itp.) programowych.

W artykule zostanie przedstawiony udział poszczególnych typów instrukcji języka źródłowego w zrealizowanych programach oraz dokonane porównanie pod tym względem trzech wykorzystywanych najczęściej języków w Polsce do przetwarzania danych takich jak PLAN, COBOL i PL/1. Ilustrujący materiał statystyczny obejmuje programy systemów informatycznych z zakresu gospodarki materiałowej jako dziedziny dość dobrze odzwierciedlającej pewne rodzaje prac o charakterze biurowym (a przynajmniej te, które najczęściej podlegają komputeryzacji).

Dla analizy języka PLAN rozpatrzono:
— 40 programów systemu SARGM (ZETO Gdynia)
— 20 programów systemu SEMO (ZETO Bydgoszcz)
dla analizy języka COBOL:
— 21 programów systemu GOMODRA (SOETO Warszawa)
— 10 programów systemu SIKOP-MERA (MERA Warszawa)
— 19 programów systemu gospodarki materiałowej (ZM „URSUS")
dla analizy języka PL/1:
— 28 programów systemu SYGMAT (ZETO Katowice).

Podstawowym, rozpatrywanym elementem powyższych języków jest instrukcja. Podział instrukcji wg typów w poszczególnych językach przedstawia tabela 1.

Udział procentowy poszczególnych typów instrukcji został policzony w stosunku do liczby wszystkich instrukcji w programie. Następnie obliczono średnią arytmetyczną $\bar{x}$ udziałów w ramach badanych języków oraz wariancję i współczynnik zmienności $V = S/\bar{x}$, stanowiący miarę względnego rozproszenia (rozrzutu) wyników. Małe wartości V (rzędu poniżej 0,50) mogą być traktowane jako symptom dość dużej jednorodności wyników i w konsekwencji jako oznaka reprezentatywności próbek pod względem badanej cechy (przy założeniu rozkładu normalnego).

Udział procentowy poszczególnych typów instrukcji przedstawia tabela 2.

Tabela 3 ilustruje ten sam materiał w sposób bardziej zagregowany. Wyróżnione są 4 grupy instrukcji:
A — instrukcje opisujące dane (typ 1)

B — instrukcje manipulacji na danych (typy 4, 5, 6 oraz częściowo 2)

C — instrukcje dzięki którym z programu źródłowego wybiera się faktycznie realizowaną wersję programu wynikowego dla określonych wartości danych zewnętrznych (typ 3 oraz częściowo 2)

D — pozostałe instrukcje (typy 7, 8).

We wszystkich językach zbliżone wartości otrzymano w grupach B (ok. 60%) i C (ok. 30%). Oznacza to, że na wszelkie manipulacje bezpośrednio dokonywane na danych (wejściowych — zewnętrznych lub roboczych — wewnętrznych) zużytkowanych jest zaledwie około 60% instrukcji w poszczególnych programach.

Tabela 1

Lp.	PLAN	COBOL	PL/1
1.	**Opis danych:** =DEFINE, =LOWER, =SET, =UPPER	FILE SECTION WORKING-STORAGE SECTION CONSTANT SECTION LINKAGE SECTION	DECLARE
2.	**Warunki:** BCC, BCHX, BCS, BCT BDX, BFP, BNG, BNZ, BPZ, BRN, BUX, BVC, BVCI, BVCR, EVS, BVSR, BXE BXGE BXL, BXU, BZE	IF	IF
3.	**Sterowanie:** a) skoki	GO TO NEXT SENTENCE	GO TO
	b) programowe i przerwania AUTO, CALL, CONT, DEL, DELTY, EXIT, OBEY, SMO, SUSAR, SUSBY, SUSDP, SUSIN, SUSMA, SUSTY, SUSWT, SDSUS, ENTER, RECAL, BRING, =CUE, =END, =ENTRY, =ERRORSEG, =FINISH, =MACRO, =OMIT, =PERMANENT, =PROGRAM, =STOP	ALTER, CALL, ENTER, EXIT, PERFORM, STOP ...SECTION (w części proced.)	BEGIN, CALL DO, END, ENTRY, PROCEDURE, RETURN, STOP, DELAY, ON, SIGNAL, WAIT
4.	**Operacje na danych:** ANDN, ANDS, ANDX, ERN, ERS, ERX, NGN, NGNC, NGS, NGSC, NGX, NGXC, ORN, ORS, ORX, ADN, ADNC, ADS, ADSC, ADX, ADXC, DVD, DVS, DVR, FAD, FDVD, FMPY, FSB, MPA, MPR, MPX, SBN, SBNC, SBS, SBSC, SBX, SBYC, SUM, CBD, CDB, FIX, FLOAT, MODE, TXL, TXU, SLA, SLC, SLL, SRA, SRAV, SRC, SRL	ADD COMPUTE DIVIDE MULTIPLY SUBSTRACT	=(Instrukcja podstawiania)
5.	**Ładowania, przesuwy:** DCH, DEX, DLA, DSA, GIVE, LDCH, LDCM, LDCT, LDEX, LDLA, LDN, LDNC, LDPL, LDSA, LDX, LDXC, LFP, LFPZ, MOVE, MVCH, NGN, NGNC, NGS, NGSC, NGX, NGXC, SFP, SFPZ, STO, STOC, STOZ, SDCLB, SDDEL, SDRD SDRDB, SDRDP, SDWR, SDWRB, SDWRI, SDWRU	MOVE DELETE SORT	DELETE REWRITE —
6.	**Operacje we/wy:** DELTY, DISP, DISTY, INPUT, INDIS, OUT, SUSTY, #MONITOR, #PAGE	ACCEPT DISPLAY READ WRITE	DISPLAY FORMAT GET, PUT, READ, WRETI

LP.	PLAN	COBOL	PL/1
7.	Komentarze: NULL #	IDENTYFICATION-DIV ENVIRONMENT--DIVIS DATA-DIVISION PROCEDURE--DIVISION CONFOGURATION-SECT PROGRAM-ID, AUTHOR INSTALATION, DATE-WRITTEN DATE-COMPILED SECURITY, REMARKS NOTE	(NULL-statement)
8.	Inne: ADLOT, BSP, BTM, DIS, FTM, PERI, REL, REW, SCR, SUSDP, SUSIN, UNL, WTM SDBSS, SDBTS, SDBUF, SDCLS, SDCRE, SDDEF, SDEND, SDEXT, SDFES, SDFSS, SDFTS, SDLAB SDWRS, #CMODE, #ELASTIC, #HMODE, #PERIPHERAL, #SWITCH, OFF, ON, OVER, TEST, PRQ, SUSWT, SDIND, SDUMP, #LIBRARY, #ORDFR, #OVERLAY, #COMPLETE	COPY SELECT...ASIGN CLOSE EXAMINE OPEN RELFASE RETURN SEEK	ALLOGATE CLOSE FREE LOCATE OPEN REVERT UNLOCK

Na opis danych przypada niewielka ilość instrukcji. Przyjęto bowiem za instrukcje opisu danych w COBOLU nazwy sekcji danych, zgadzając się na to, że będą one miały bardzo rozbudowany charakter.
Usprawiedliwiając takie postępowanie należy zauważyć, że podobnie rozbudowaną strukturę mają instrukcje deklaratywne PL/1, z tym że w przeciwieństwie do COBOLU występować one mogą w dowolnym miejscu programu.

Tabela 2

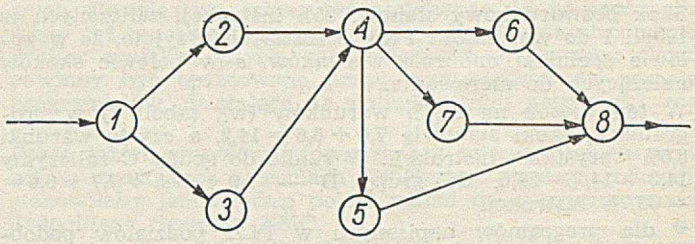
Rodzaj instrukcji	PLAN $\bar{x}$ V	COBOL $\bar{x}$ V	PL/1 $\bar{x}$ V
Długość programu	388—0,57	347—0,99	326—0,74
1. Dane	1,4—1,07	1,1—0,73	9,3—0,49
2. Warunki	20,7—0,30	13,7—0,36	9,7—0,26
3. Sterowanie	14,3—0,48	27,0—0,36	25,9—0,24
4. Operacje	7,9—0,61	6,7—0,88	30,0—0,37
5. Ładowania	40,5—0,26	33,7—0,31	—
6. we/wy	7,7—0,68	10,7—0,64	22,9—0,038
7. Komentarze	8,0—0,79	4,8—0,69	2,3—0,83
8. Inne		2,5—1,04	
Etykiety	17,3—0,29	16,9—0,34	10,5—0,45

Dla grupy A wydaje się w każdym razie możliwa do przyjęcia teza, że nie przekraczają one kilku procent wszystkich instrukcji programów.
W rezultacie otrzymujemy, że na pewno ok. 60—65% instrukcji poświęconych jest na czynności rzeczywiście implikowane przez zastosowanie biurowe. Jaki charakter ma pozostałych co najmniej 35% instrukcji?
Najwięcej z nich przypada na grupę C (sterowanie i część warunków) — 30% (a w COBOLU ok. 35%). Reszta zaś to komentarze lub bezpośrednia współpraca z urządzeniami (konieczność jawnego uwzględnienia w programie charakterystyki sprzętu).
Instrukcje grupy C nie są bezpośrednio uzasadnione zastosowaniami. Aby się o tym przekonać rozpatrzymy następujący model przetwarzania danych (zbliżony do rzeczywistego); praca ich będzie tak zorganizowana, że przetwarzający (pracownik lub grupa osób) nie musi przerywać jakiejś, właśnie wykonywanej grupy czynności dla uzyskania danych, które jeszcze nie są wyliczone (przez innego lub tego samego przetwarzającego).
Nie uważamy przy tym za przerwanie konieczność znalezienia potrzebnych danych lecz jedynie konieczność przetworzenia danych w międzyczasie, gdy okazują się one niezbędne.

Tabela 3

GRUPA INSTRUKCJA	PLAN	COBOL	PL/1
A	1,4	1,1	9,3
B	62,7	57,3	57,7
C	28,5	34,5	30,7
D	8,0	7,3	2,3

W zespole, jako całości, mamy więc podział pracy na poszczególne procedury dokonywane w sposób ciągły, a więc bez przerwań na innego typu czynności. Procedury te układają się w tym zespole w sieć typu PERT: między poszczególnymi węzłami przedstawiającymi procedury istnieją zależności czasowe typu równoległości, uprzedniości lub następstwa działań.



Rys. 1. Schemat sieci typu PERT. O — oznacza różne czynności do wykonania w ramach całego zestawu przedstawionego siecią, → — oznacza kierunek przepływu wyników jednej czynności jako niezbędnych do wykonania czynności kolejnej

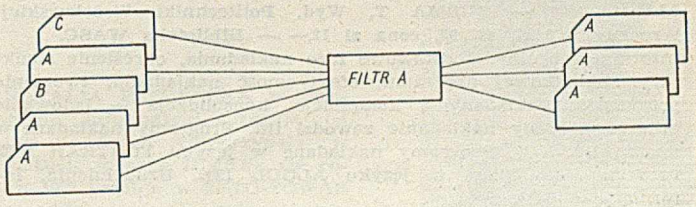
Nałóżmy teraz na taki model technikę komputerowego przetwarzania danych. Wyżej określonym procedurom odpowiadają w tej technice programy (tak przynajmniej było we wszystkich analizowanych systemach).

Gdyby praca komputera, tzn. wykonanie programu miało być tak samo zorganizowane jak w wyżej opisanym modelu, to w ramach programu istniałyby kolejne przejścia od instrukcji do instrukcji w sposób sekwencyjny bez konieczności przeskoków, powrotów itp. Tego typu przetwarzanie komputerowe jest możliwe, np. w trybie konwersacyjnym lub przy zastosowaniu techniki programowania modularnego.

Jednakże w większości przypadków (tak było w omawianych systemach) stosuje się tryb programowania, w którym program źródłowy umożliwia realizację różnych sekwencji instrukcji w każdym konkretnym przypadku. Możliwości te realizowane są przez instrukcje sterowania i niektóre warunki. Tego typu instrukcje nie są więc bezpośrednio implikowane przez zastosowania.

Zanalizujmy, kiedy warunki są bezpośrednio (a więc również w ramach przedstawionego wyżej modelu), a kiedy pośrednio (poprzez przyjęty algorytm) implikowane zastosowaniami. Aby odpowiedzieć na to pytanie zastanówmy się jakie funkcje spełniają warunki w programach komputerowych. Pełnią one dwie różne role. Mogą występować jako:

- FILTR wybiorczy jednostek danych (np. zapisów ze zbioru), na których ma nastąpić operacja



Rys. 2. Schemat działania FILTRU

● DYSTRYBUTOR dróg w algorytmie. Działa na zmiennych roboczych lub ze zbioru danych i każdorazowo powoduje wybór ciągu instrukcji do przetwarzania z listy instrukcji umieszczonych w programie.

Do funkcji typu FILTR zaliczona została także funkcja kontrolna warunku, wówczas gdy chodzi o sprawdzenie, czy dana ze zbioru spełnia narzucone jej ograniczenia (co do postaci, zakresu itp.). Podyktowane to zostało faktem, że kontrola polega na przepuszczeniu do dalszego przetwarzania a więc FILTR-owaniu danych spełniających zadane warunki.

Warunki z funkcji FILTR należy więc uznać za implikowane bezpośrednio przez zastosowanie, a w funkcji DYSTRYBUTOR za implikowane pośrednio.

W rozpatrywanych systemach sytuacja była następująca:

● w programach opartych na języku COBOL instrukcja JF była w 45% określona na danych nie roboczych (plik typu INPUT lub INPUT-OUTPUT), a więc było co najmniej 55% wystąpień w roli DYSTRYBUTORA (co najmniej, bo może się przecież zdarzyć, że także DYSTRYBUTOR działał będzie na danej ze zbioru danych). W ten sposób do grupy C zaliczymy: 27,0 + 7,5 = 34,5, a do B: 6,7 + 33,7 + 10,7 + 6,2 = 57,3% instrukcji

● w programach opartych na języku PLAN jest rzeczą niemożliwą (poza instrukcją BRN — skok bezwarunkowy) oddzielić warunek od skoku, ponieważ są one związane jedną instrukcją.

Skok bezwarunkowy stanowi 35% instrukcji zaliczonych w tabeli 1 do warunków. Przy pozostałych przyjęto, że w połowie spełniają one rolę warunków, a w połowie skoków należących do sterowania.

W ten sposób na 20,7% warunków (wg tabeli 1) w programach, skoki stanowią 7,6 + 6,6 = 14,2, a czyste warunki 6,6% wszystkich instrukcji. W sumie do grupy C zaliczymy 14,3 + 14,2 = 28,5, do grupy B — 7,9 + 40,5 + 7,7 + 6,6 = = 62,7% instrukcji

● dla programów napisanych w PL/1 podziałów podobnych jak w COBOLU nie przeprowadzono. Wydaje się jednak uzasadnione przyjęcie, że około 50% rozkazów IF występuje w roli FILTRÓW, a pozostałe 50% w roli DYSTRYBUTORA.

```
START;
ZMIENNA ROBOCZA X;
. . .
X:=A+5;
. . .
IF X < 10 THEN GO TO E1
          ELSE
          IF X < 20 THEN GO TO E2
          ELSE GO TO E4;
. . .
. . .
E1: . . . . .
. . .
E2: . . .
. . .
E3: . . .
. . .
E4: . . .
. . .
KONIEC;
```

Rys. 3. Schemat działania instrukcji IF jako DYSTRYBUTORA

Otrzymamy więc dla PL/1:
grupa C — 25,9 + 4,8 = 30,7
grupa B — 30,0 + 22,9 + 4,8 = 57,7

Wyniki powyższe ukazują jak duży ponad 30% balast pojęciowy niosą ze sobą różne języki spośród badanych — balast o zbliżonym poziomie, mimo, że są wśród nich języki różnych rzędów lub technika ich użycia w stosunku do wymagań stawianych przez zastosowania.

Dalsze analizy zebranego materiału mogą dostarczyć ciekawych wniosków jeśli chodzi np. o stopień wykorzystania potencjału języków, wyodrębnienie pewnych „szkół" programowania, sposób szkolenia programistów, itp.

PRZEGLĄD WYDAWNICTW

Bibliografia wydawnictw polskich z dziedziny informatyki

● **Laboratorium cyfrowych układów automatyki** — BARGIELSKI M. Wyd. Politechniki Śląskiej im. W. Pstrowskiego. Gliwice 1975, ss. 83, cena zł 10.— Skrypty uczelniane nr 558.

Materiał zawarty w skrypcie obejmuje opisy stanowisk laboratoryjnych, programy ćwiczeń ze wskazówkami odnośnie opracowywania sprawozdań oraz zagadnienia teoretyczne związane z poszczególnymi ćwiczeniami. Omówiono następujące ćwiczenia: organizacja maszyny cyfrowej, arytmometr, struktura minikomputera, cyfrowe układy scalone, układy zapisu i odczytu pamięci bębnowej, konwertory A/C i C/A, dynamika układów cyfrowych, generator binarnego sygnału pseudoprzypadkowego.

Skrypt został opracowany jako pomoc naukowa dla studentów wykonujących ćwiczenia laboratoryjne z cyfrowych układów automatyki.

● **Programy nakładane w językach PLAN 3, FORTRAN 1900, ALGOL 1900** — SURMA T. Wyd. Politechniki Wrocławskiej, Wrocław 1976 r., ss. 92, cena zł 12.— — Biblioteka WASC.

Informacje ogólne: podstawowe idee nakładania, określenie struktury nakładkowej programu, efektywność nakładania, tworzenie programów nakładanych, kompilacja, konsolidacja — ładowanie programu, kiedy nakładanie zawodzi itd. Programy nakładane w języku PLAN 3. Programy nakładane w języku FORTRAN 1900. Programy nakładane w języku ALGOL 1900. Uzupełnienia, informacje dodatkowe.

Materiały przeznaczone są dla programistów i studentów wyższych szkół technicznych.

Układy scalone — KÜHN E., SCHMIED H. Tłum. wyd. niem. z 1972 r. Wyd. Komunikacji i Łączności, Warszawa 1976, s. 190, cena zł 25.—

Wiadomości ogólne: serie układów TTL, bramka NANP, przerzutnik RS, JK, D. Parametry statyczne układów TTL. Parametry dynamiczne układów TTL. Wpływ temperatury i napięcia zasilania. Wpływ zakłóceń. Niezawodność. Dopasowanie układów TTL do obciążeń zewnętrznych. Układy specjalne. Multiwibratory. Rozdzielacze. Dzielniki częstotliwości. Liczniki. Dekodery. Dodatki. Książka zawiera podstawowe wiadomości potrzebne do projektowania układów cyfrowych zawierających układy TTL. Przeznaczona jest dla inżynierów i techników elektroników.

System operacyjny GEORGE 3 w procesie przetwarzania informacji — BIELENIK E. i inni, Wyd. Ośrodka Badawczo-Rozwojowego Informatyki INFORNA, Warszawa 1976, s. 161, cena zł 58.— **Problemy Informatyki.**

Proces przetwarzania informacji jako proces technologiczny. Rozwój systemów cyfrowych. Rozwój oprogramowania operacyjnego serii ODRA 1300. Koncepcja systemu operacyjnego GEORGE 3. Pamięć systemu GEORGE 3. Opisy zadań w GEORGE 3. Elementy organizacyjne systemu. Problemy wdrażania systemu GEORGE 3 w ośrodku obliczeniowym. Słownik polskich terminów systemu operacyjnego GEORGE 3. Materiały przeznaczone są dla programistów.

Oprac. A K.

ANNA PASTUSIAK
Instytut Informacji Naukowej, Technicznej i Ekonomicznej
NATALIA PECZYŃSKA
TERESA RUMIŃSKA
ZETO ZOWAR
Warszawa

Adaptacja pakietów TEXTPAC i STAIRS w informatycznych systemach dokumentacyjnych

PRZYCZYNY I WARUNKI ADAPTACJI PAKIETÓW PROBLEMOWYCH

Dynamiczny rozwój zastosowań komputerów do problemów wyszukiwania informacji tekstowej i wydawnictw informacyjnych (biuletynów, zestawień, dokumentacyjnych indeksów) jest stymulatorem rozwoju szeregu działów informatyki rozwiązujących teoretycznie i praktycznie problemy: języków informacyjnych, automatycznego indeksowania i przekładu maszynowego, formatu i struktur danych, organizacji i sterowania dostępem do bardzo wielkich zbiorów danych (baz danych), optymalizacji algorytmów wyszukiwania informacji, typizacji programów wydawniczych, teletransmisji, dostosowania wyszukiwania bezpośredniego (*on-line*) do możliwości osób nieobeznanych z językami programowania i komputerami itp. Z kolei rozwój tych kierunków działalności powinien być uwzględniony przy realizacji poszczególnych funkcji projektu zautomatyzowanego systemu informacyjnego — zdefiniowanych w zadaniu projektowym. Czynnikiem limitującym zakres włączania aktualnych osiągnięć wymienionych wyżej działów informatyki, stanowić powinny jedynie dostępne środki realizacji systemu — sprzętowe i ludzkie.

Wśród kompleksu zagadnień wymagających w projekcie ścisłego ustalenia, takich jak np. organizacja spływu dokumentów źródłowych, zapewnienie wprowadzenia w dokumentach źródłowych danych niezbędnych do uzyskania pożądanych serwisów informacyjnych, dostosowanie formatu dokumentów wejściowych (tam, gdzie to jest możliwe) do wybranej wersji przenoszenia danych na nośniki maszynowe, usprawnienie obiegu dokumentów, organizacja zbiorów informacyjnych, zabezpieczenie danych, wybór i przygotowanie języka informacyjnego, zdefiniowanie sposobu indeksowania, określenie postaci wyników i form rozprowadzenia informacji wynikowych itd., szczególne miejsce zajmuje problem oprogramowania procesów przetwarzania informacji.

Algorytmy wyszukiwania informacji tekstowej z dużych zbiorów informacyjnych należą do najbardziej złożonych i pracochłonnych zadań programowania i wymagają (w racjonalnych terminach realizacji oprogramowania projektu, a więc nie przekraczających jednego roku) licznej grupy wysoko kwalifikowanych programistów, wykorzystujących w trakcie testowania programów znaczny limit godzin pracy komputera wyposażonego w dużą pamięć operacyjną. Wobec ograniczonych kadr odpowiednich fachowców i skromnych praktycznie możliwości korzystania z czasu komputerów w celach eksperymentalnych (kolidujących często z potrzebami eksploatacji użytkowej), najbardziej racjonalną drogą zastosowania komputerów w systemach informacyjnych wydaje się wykorzystanie gotowych pakietów użytkowych, dostępnych odpłatnie, a czasem nawet nieodpłatnie.

Nie oznacza to bynajmniej negowania potrzeby rozwijania w Polsce rozwiązań programowych dla problemów typowych. Rozwiązania takie, ukierunkowane na rozpowszechniony już znacznie w kraju, a szerzej jeszcze we wspólnocie krajów socjalistycznych, sprzęt Jednolitego Systemu EMC, powinny być opracowane przez wyspecjalizowane zespoły programistów, działające na potrzeby całej gospodarki narodowej.

Praca takich zespołów, prowadzona z wykorzystaniem wyników uzyskanych w ramach współpracy międzynarodowej, stanowiłaby jednocześnie polski wkład do oprogramowania Jednolitego Systemu EMC.

Do czasu stworzenia odpowiednio bogatego oprogramowania tego systemu, warto wykorzystać istniejące już pakiety użytkowe, zwłaszcza zorientowane problemowo, tj. obejmujące cały kompleks funkcji ukierunkowanych na zastosowanie w określonej dziedzinie.

Różnorodność klasyfikacji pakietów i niejednolitość opisu samych pakietów sprawia, że dobór właściwego pakietu wymaga konsultacji wyspecjalizowanego ośrodka, mającego dobre rozeznanie rynku oprogramowania oraz dysponującego kadrą zdolną do merytorycznej oceny funkcji pakietu, opłacalności jego nabycia i możliwości jego adaptacji w konkretnych warunkach technicznych użytkownika.

Poradnictwo tego rodzaju można uzyskać na terenie kraju głównie w sieci ZETO, a w Warszawie w OBRI, ZOWAR, IDKKAP oraz IINTE.

Warto podkreślić, że adaptacja pakietu nie oznacza na ogół pełnego pozbycia się problemów związanych z oprogramowaniem systemu.

Poza oczywistą funkcją sprawdzenia działania pakietu zgodnie z jego deklarowanymi możliwościami, zachodzi na ogół konieczność dostosowania programów do specyficznych potrzeb użytkownika.

Najczęściej będzie to zadanie dostosowania danych źródłowych do formatu wejściowego pakietu oraz przekształcenia formatu danych wynikowych do postaci pożądanej przez użytkownika. Czasem wyłania się również możliwość wzbogacenia funkcji wykorzystywanego pakietu kosztem pewnych zmian w programach.

Mgr inż. ANNA PASTUSIAK ukończyła Politechnikę Warszawską w 1961 r. Zagadnieniami automatyzacji procesów informacyjnych zajmuje się od 1971 r., początkowo w Centralnym Ośrodku Informacji Patentowej — U.P. PRL, a od 1973 r. w Zakładzie Systemów Informacyjnych IINTE.

NATALIA PECZYŃSKA w dziedzinie informatyki pracuje od 1964 r. Pracę rozpoczynała w CODKK (obecne IDKKP). Od roku 1972 pracuje w ZETO-ZOWAR w dziale wyszukiwania informacji.

Wprowadzenie nawet niewielkich zmian do programów — nie zawsze zresztą jest formalnie dopuszczalne (twórca pakietu nie bierze wówczas najczęściej odpowiedzialności za prawidłowe działanie pakietu) — wymaga umiejętności „rozpakowywania" programów i zorientowania się w ich logice, a więc biegłej znajomości zarówno języków programowania jak i systemu operacyjnego, pod którego kontrolą programy te działają.

Jeżeli jednak dysponuje się odpowiednim sprzętem i kadrą, to warto zdecydować się na tak szeroko pojętą adaptację, zwłaszcza, że przeróbki programów, dzięki konieczności poznania nieraz bardzo finezyjnych rozwiązań programowych zastosowanych w pakietach, stają się jednocześnie szkołą zaawansowanych metod programowania i przyczyniają się do podniesienia kwalifikacji zespołu adaptującego. Podpatrywane rozwiązania mogą być również źródłem pomysłów przy kolejnych własnych rozwiązaniach programowych. Potwierdza to nasze opisane dalej doświadczenie.

WYBÓR PAKIETU I OKREŚLENIE ZAKRESU ZMIAN I UZUPEŁNIEŃ PROGRAMÓW

Implementacja zautomatyzowanego systemu informacyjnego o pracach naukowo-badawczych, zrealizowana w Instytucie Informacji Naukowej, Technicznej i Ekonomicznej, stanowi dość poglądowy przykład zalet wynikających z adaptacji odpowiedniego pakietu programów, ilustrując jednocześnie zakres prac towarzyszących takiej adaptacji.

Z założenia system informacji o pracach naukowo-badawczych miał za zadanie:

- emitować bogaty serwis wydawniczy w postaci biuletynu informacyjnego ze spisem treści i towarzyszącymi indeksami: autorskim, przedmiotowym, instytucji realizujących, autorów prac doktorskich, promotorów, autorów prac habilitacyjnych (indeksy te miały być dodatkowo kumulowane w okresach rocznych i ewentualnie wieloletnich)
- umożliwiać selektywne rozpowszechnianie informacji bieżącej, tj. odpowiedzi na stałe pytania abonentów (tzw. profile SDI)
- zabezpieczyć retrospektywne wyszukiwanie informacji według dowolnego elementu opisów prac wprowadzanych do założonej i utrzymywanej bazy danych systemu.

Wobec potencjalnie szerokiego zapotrzebowania na takie systemy przy automatyzacji funkcji informacyjnych bibliotek i ośrodków informacji starano się przedstawić przebieg realizacji projektu w sposób przystępny dla pracowników służby inte mniej obeznanych z problemami programowania.

Spośród przeanalizowanych przez nas pakietów programowych łączących funkcje edycyjne (tj. zapewniających określone postacie wydruków, a przede wszystkim umożliwiających łatwą i wielokrotną korektę zapisu danych) z funkcjami wyszukiwawczymi wybraliśmy pakiet TEXTPAC.

O wyborze, oprócz samych funkcji realizowanych przez programy pakietu, zdecydował fakt, że pakiet ten dostosowany jest do konfiguracji komputera, która mogła być użyta do eksploatacji systemu, a ponadto — co nie jest sprawą błahą — korzystanie z tego pakietu jest nieodpłatne.

TEXTPAC [1] jest bowiem pakietem programów z biblioteki programów użytkowych IBM, za działanie których firma nie ponosi odpowiedzialności. Daje to dodatkowo swobodę „bezkarnego" wprowadzania zmian w programach.

[1] TEXTPAC — S/360 Normal Text Information Processing, Retrieval, and Current Information Selection System. 360D-06.7.020

Mgr inż. TERESA RUMIŃSKA w 1960 r. ukończyła Politechnikę Warszawską, wydział Komunikacji. Od 1 lutego 1966 r. pracuje w Zakładzie Elektronicznej Techniki Obliczeniowej ZOWAR, obecnie na stanowisku kierownika działu Wyszukiwania Informacji.

TEXTPAC przeznaczony jest do wydawnictw i wyszukiwania informacji w tekstach zapisanych w języku naturalnym w oparciu o częstotliwość występowania słów w tekście oraz o logiczne pozycyjne powiązania między słowami.

Pakiet obejmuje 30 programów umożliwiających użytkownikom:

- zakładanie i utrzymywanie podstawowego zbioru informacyjnego zawierającego pełną treść dokumentów (w tym przypadku dokumentem był opis pracy naukowo-badawczej w zakresie typowej karty dokumentacyjnej)
- wyszukiwanie selektywne (profilowe) w trybie zestawienia bieżącej, tj. stosunkowo niewielkiej liczby danych ze znaczną liczbą pytań
- wyszukiwanie retrospektywne w trybie zestawienia ograniczonej liczby pytań z dużą liczbą dokumentów (bazą danych)
- wydawanie biuletynu informacyjnego (wydruk pełnego lub skróconego tekstu dokumentów) oraz 5 rodzajów indeksów.

Jednakże wydawnictwa przewidywane przez TEXTPAC nie spełniały wszystkich wymagań użytkownika. Użytkownik systemu wymagał bowiem aby układ biuletynu opierał się na wprowadzonej do opisu dokumentu klasyfikacji Międzynarodowego Centrum Informacji Naukowej i Technicznej (MCINT) i to w sposób niesekwencyjny, ponadto liczba żądanych indeksów była większa niż przewiduje to pakiet TEXTPAC. Również wymagany zakres kontroli występowania i formatu poszczególnych danych był znacznie szerszy niż w TEXTPAC.

Z analizy działania programów nasunął się ponadto wniosek, że z racji sekwencyjnego charakteru przeszukiwania zbioru podstawowego i braku możliwości interakcyjnego zmieniania pytań (konwersacji), wyszukiwania retrospektywne nie będą efektywnie realizowane przez ten pakiet. W związku z tym zdecydowano się:

- maksymalnie wykorzystać elementy pakietu związane z zakładaniem i utrzymywaniem zbioru podstawowego oraz wydawnictwem biuletynu i indeksów
- zachować pakietowe rozwiązanie wyszukiwań selektywnych
- do wyszukiwań retrospektywnych wykorzystać pakiet STAIRS [2] umożliwiający wyszukiwanie w trybie konwersacyjnym. Należy wyjaśnić, że STAIRS, będący odpłatnym pakietem z biblioteki programów użytkowych IBM, wykorzystuje na wejściu zbiór w formacie generowanym przez pakiet TEXTPAC.

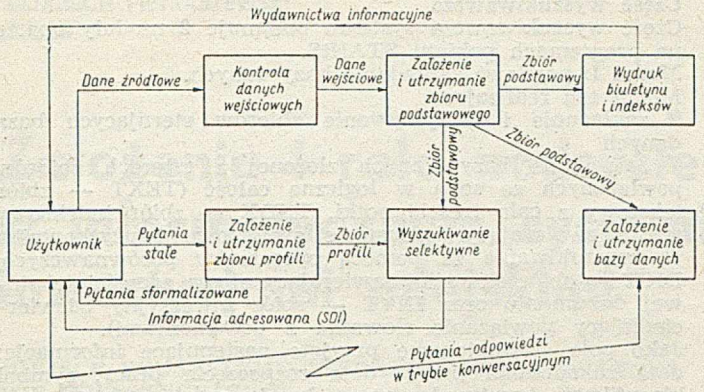
W celu zapewnienia
— odpowiedniej kontroli danych
— pożądanego układu biuletynu
— uzyskania powiązań między zbliżonymi tematycznie dokumentami
— generowania wszystkich potrzebnych indeksów, a także minimalizacji prac związanych z procesami przeniesienia danych z dokumentu źródłowego na nośnik maszynowy oraz uniknięcia (do czasu ustalenia ostatecznej postaci notki w biuletynie) pisania własnego programu wydruku biuletynu, wykonano następujące prace adaptacyjne:

- opracowano formularz karty informacyjnej pracy naukowo-badawczej przystosowany do bezinstrukcyjnego przenoszenia danych na nośnik maszynowy. Kody pól danych wydrukowane na formularzu gwarantowały uzyskanie odpowiedniego układu biuletynu i wykorzystania niektórych danych jako elementów indeksów
- opracowano programy kontroli formalnej i logicznej danych wejściowych (z kart informacyjnych) i konwersji tych danych na format wymagany przez pakiet TEXTPAC
- rozbudowano część wydawniczą systemu o programy wydruku dodatkowych indeksów i spisu treści biuletynu w oparciu o założoną kartotekę klasyfikacji MCINT.

OPIS STRUKTURY, FUNKCJI I KONFIGURACJI KOMPUTEROWEJ SYSTEMU

Strukturę systemu wydawniczo-wyszukiwawczego informacji o pracach badawczo-rozwojowych, stanowiącego połączenie pakietu TEXTPAC z oprogramowaniem uzupełniającym oraz pakietu STAIRS do wyszukiwania retrospektywnego z możliwością prowadzenia zdalnej konwersacji z bazą danych pokazuje rys. 1.

[2] STAIRS — Storage and Information Retrieval System. 5734-X23.

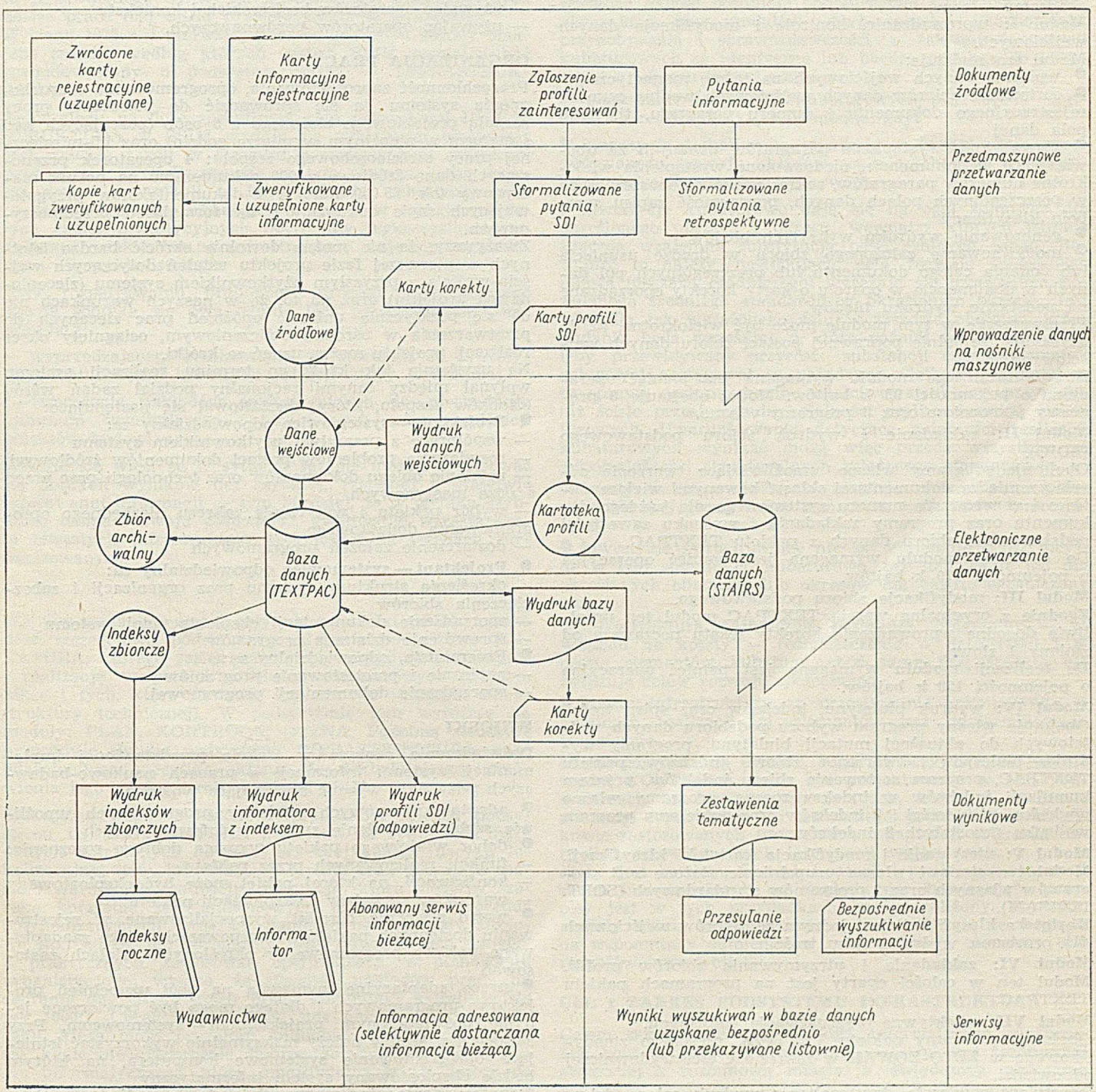


Rys. 1. Struktura systemu

Powiązanie dokumentów i głównych zbiorów w systemie przedstawia rys. 2.

Eksploatacja systemu odbywać się może na komputerach IBM 360 lub IBM 370 w systemie operacyjnym OS z wielkością obszaru pamięci operacyjnej 240 kbajtów. Pod kontrolą pakietu sterującego teletransmisją CICS [3], zapewniającego obsługę zarządzania: zadaniami, terminalami, zbiorami, programami, pamięcią.
Wymagana konfiguracja urządzeń zewnętrznych:
— 2 jednostki pamięci taśmowej
— czytnik kart
— drukarka wierszowa
— jednostki pamięci dyskowej — ilość zmienna począwszy od jednej jednostki
— terminale typu IBM 3270 u użytkownika oraz terminal IBM 2740 w ośrodku EPD jako terminal główny.
Cykl przetwarzania danych przy użyciu partii ok. 1800 dokumentów (obejmujący wprowadzenie danych do bazy danych oraz emisję wydruków zbioru podstawowego, biuletynu i indeksów), zajmuje na komputerze IBM 360/50 ok. 2 godzin.

[3]) Customer Information Control System nr 5736, U11 lub 5734-XX7.



Rys. 2. Schemat powiązań dokumentów i głównych zbiorów danych

Czas wyszukiwania w trybie konwersacyjnym zależy od ilości sesji terminalowych.

OPIS OPROGRAMOWANIA SYSTEMU

Informatyczna część systemu obejmuje przetwarzanie partiowe w części „edycyjnej" opartej na pakiecie TEXTPAC oraz przetwarzanie partiowe i konwersacyjne w części „wyszukiwawczej" opartej na pakiecie STAIRS.

Część edycyjna

Edycyjna część przetwarzania da się ująć w 7 modułów wykorzystujących ogółem 14 programów z pakietu TEXTPAC, 4 programy standardowe (utilities IBM/360) oraz 7 programów własnych. 3 spośród zastosowanych programów TEXTPAC przyjęte zostały w wersji uprzednio zmodyfikowanej przez ZETO-ZOWAR.
Modyfikacje w tych programach umożliwiły rozszerzenie pól indeksowych oraz wprowadzanie tekstów zapisanych jedynie dużymi literami bez specjalnych wyróżników.
Ze względu na zachowanie niezbędnej w artykule zwięzłości opis modułów ograniczony jest jedynie do określenia realizowanych przez nie funkcji i podanie wymagań odnośnie do pamięci głównej w modułach zawierających programy własne.
Moduł I: wprowadzanie, kontrola i modyfikacja danych wejściowych.
Moduł ten realizuje:
- wpisanie danych wejściowych na taśmę magnetyczną
- sortowanie zbiorów danych wejściowych według numeru rejestracyjnego dokumentu i numeru paragrafu, tj. kodu pola danej
- sprawdzanie wystąpienia paragrafów uznanych za obowiązujące w dokumencie, niedozwolone wystąpienia wielokrotne numerów paragrafów, zakres znaków dopuszczalnych w poszczególnych polach danych, poprawność zapisu symbolu klasyfikacji
- generowanie wydruku wykrytych błędów
- modyfikowanie założonego zbioru w drodze usunięcia lub dodania całego dokumentu lub poszczególnych pól danych w dokumencie w oparciu o karty korekty sporządzane na podstawie wydruku błędów.

Przetwarzanie w tym module może być wielokrotne, aż do uzyskania formalnie poprawnej postaci zbioru danych wejściowych.
Do realizacji tego modułu wymagana jest pamięć operacyjna o pojemności 95 k bajtów. Moduł obejmuje 3 programy pomocnicze oraz 3 programy własne.
Moduł II: zakładanie i wydruk zbioru podstawowego (EDIO).
Obejmuje programy własne, umożliwiające tworzenie odsyłaczy między dokumentami sklasyfikowanymi wieloaspektowo i generowanie numeru systemowego dla każdego dokumentu oraz programy zakładania i wydruku zawartości podstawowego zbioru danych z pakietu TEXTPAC.
Do realizacji modułu wymagana jest pamięć operacyjna o pojemności 170 k bajtów.
Moduł III: modyfikacja zbioru podstawowego.
Zgodnie z oryginalną wersją TEXTPAC moduł ten umożliwia dogodne wprowadzenie korekty tekstu począwszy od zmiany 1 słowa.
Do realizacji modułu wymagana jest pamięć operacyjna o pojemności 128 k bajtów.
Moduł IV: wydruk biuletynu, indeksów oraz spisu treści.
Obejmuje własny program wyboru podzbioru danych wejściowych do aktualnej mutacji biuletynu, programy wydruku biuletynu i tworzenia zbioru indeksów, pakietu TEXTPAC, program sortowania zbioru indeksów, program kumulacji indeksów w indeksy roczne, programy własne wydruku spisu treści i 3 indeksów oraz pakietowy program wydruku pozostałych 3 indeksów.
Moduł V: utrzymanie i modyfikacja kartoteki klasyfikacji.
Moduł ten stanowi własne uzupełnienie złożone z 2 programów własnych oraz 2 programów standardowych (SORT, IEBISAM).
Kartoteka klasyfikacji jest jednym ze zbiorów wejściowych dla programu wydruku spisu treści.
Moduł VI: zakładanie i utrzymywanie zbiorów profili.
Moduł ten w całości oparty jest na programach pakietu TEXTPAC.
Moduł VII: selektywne wyszukiwanie informacji.
Obejmuje programy pakietu TEXTPAC częściowo zmodyfikowane w ZETO-ZOWAR w zakresie części wydawniczej odpowiedzi.
Odpowiedź zawierała dwuszpaltowy wydruk całego dokumentu bez ograniczeń istniejących w wersji oryginalnej.

Część wyszukiwawcza

Część wyszukiwawcza systemu obejmuje 2 moduły oparte na programach pakietu STAIRS.
Moduł I: partiowe zakładanie bazy danych.
Moduł ten realizuje:
- zakładanie i utrzymywanie zbiorów sterujących bazą danych
- zakładanie bazy danych złożonej z czterech zbiorów powiązanych ze sobą w logiczną całość (TEXT — zbiór tekstowy z całą treścią notki, TNDX — zbiór indeksowy tekstu zawierający informacje systemowe i użytkownika do wyszukiwania za pomocą operatorów porównawczych, DICT — zbiór słownika zawierający słowa z części tekstowej dokumentu oraz INVT — zbiór inwersyjny odzwierciedlający powiązanie słownika z dokumentami).

Jako pole sformatowane przyjęto następujące informacje: data zakończenia pracy, data rozpoczęcia pracy, symbol klasyfikacji, koszty planowane, koszty rzeczywiste
- utrzymywanie bazy danych — łączenie poszczególnych baz w jedną całość tzn. retrospektywną bazę danych.

Moduł II: konwersacyjny dostęp do bazy danych.
STAIRS jest pakietem przechowywania i wyszukiwania informacji tekstowej pozwalającym użytkownikowi formułować pytania w dwojaki sposób:
— używając operatorów boolowskich i kontekstu
— używając operatorów porównawczych.

ORGANIZACJA PRACY

Pracochłonność zaprojektowania, oprogramowania i uruchomienia systemu da się sprowadzić do półrocznej pracy zespołu projektowego złożonego z 3 osób (pracujących nad systemem w niepełnym wymiarze godzin) oraz 1,5-miesięcznej pracy sześcioosobowego zespołu: 4 operatorek przenoszących dane źródłowe z 600 dokumentów na nośnik maszynowy (ok. 25 000 kart) i 2 dokumentalistów przygotowujących zapis dokumentów na formularzu przedmaszynowym.
Zważywszy, że nie można dowolnie skrócić bardzo istotnych w pierwszej fazie projektu ustaleń dotyczących wejścia-wyjścia z przyszłym użytkownikiem systemu (zleceniodawcą projektu) oraz na to, że w naszych warunkach nie da się praktycznie uniknąć opóźnień prac zlecanych do przetwarzania w ośrodku obliczeniowym, osiągnięty okres realizacji projektu można uznać za krótki.
Na uzyskanie tak krótkiego terminu realizacji systemu wpłynął między innymi racjonalny podział zadań wśród członków zespołu, który ukształtował się następująco:
- **Projektant — systemowiec** odpowiedzialny za:

— współpracę z przyszłym użytkownikiem systemu
— rozwiązanie problemów postaci dokumentów źródłowych
— ustalenie obiegu dokumentów oraz technologii prac przed i poza maszynowych
— wybór pakietu i wytyczenie zakresu niezbędnego oprogramowania dodatkowego
— dostarczenie założeń programowych

- **Projektant — systemowiec** odpowiedzialny za:

— określenie struktury systemu oraz organizacji i zabezpieczenia zbiorów
— sporządzenie dokumentacji eksploatacyjnej systemu
— sprawdzenie działania programów

- **Programista,** odpowiedzialny za:

— napisanie i przetestowanie programów
— sporządzenie dokumentacji programowej.

WNIOSKI

Doświadczenia zdobyte w trakcie przedstawionej implementacji systemu informacji o pracach naukowo-badawczych pozwalają wysnuć następujące wnioski:
- adaptacja firmowych pakietów programowych umożliwia szybkie wdrożenie systemów informatycznych
- dobór właściwego pakietu wymaga dobrego rozeznania:

— funkcji realizowanych przez pakiet
— konfiguracji, na której pakiet może być eksploatowany
— warunków sprzedaży i eksploatacji pakietu

- warto stworzyć ośrodki wyspecjalizowane w rejestrowaniu dostępnych pakietów i rozpowszechniające zaadaptowane pakiety programowe w określonych działach zastosowań
- prace adaptacyjne wymagają na ogół uzupełnień projektowo-programowych i dlatego mogą być powierzone jedynie doświadczonym projektantom — systemowcom. Przy adaptacji pakietów należy maksymalnie wykorzystać istniejące oprogramowanie systemowe komputera, na którym będzie eksploatowany system informatyczny
- prace adaptacyjne przyczyniają się do szybszego podnoszenia kwalifikacji programistów.

WOJCIECH PIETRASZEWSKI
Kraków

Założenia i rola podsystemu „INFRASTRUKTURA" w informatycznym systemie sterowania inwestycjami miejskimi

MIEJSCE PODSYSTEMU INFRASTRUKTURA W SYSTEMIE INWESTYCYJNYM

W numerze 7—8/75 INFORMATYKI opublikowany został artykuł opisujący informatyczny System Sterowania Inwestycjami Miejskimi (SSIM), który był wówczas projektowany w zespole Zakładu Systemów Regionalnych Oddziału Krakowskiego OBRI. Na skutek zmian organizacyjnych dalsze prace nad SSIM przejęło ZETO w Krakowie.
W ciągu 1975 r. i pierwszej połowy 1976 roku zostały ustalone zasady, według których system SSIM powinien być rozbudowywany o podsystem SSIM — INFRASTRUKTURA.

Ze względu na dość specyficzną rolę podsystemu INFRASTRUKTURA w SSIM wydaje się właściwym omówienie jego założeń i roli w sterowaniu inwestycjami miejskimi przy użyciu ETO. Przemawia za tym również coraz większy wpływ inwestycji w infrastrukturę techniczną miast na tempo i koszty urbanizacji kraju, a w sytuacji napiętych zadań inwestycyjnych optymalne wykorzystanie ponoszonych nakładów na infrastrukturę staje się zagadnieniem niezwykle istotnym.

Przy rozpatrywaniu problemów procesu inwestycyjnego miasta spotyka się dwa istotne lecz przeciwstawne sobie postulaty:

— wyprzedzającego realizowania (w stosunku do inwestycji kubaturowych) inwestycji infrastruktury technicznej

— zamrożenia nakładów inwestycyjnych do niezbędnego minimum wynikającego z technicznych praw realizacji obiektów budowlanych.

Wydaje się więc, że w sytuacji tej nie można spodziewać się dobrych wyników bez szybkiej, rzetelnej i właściwie skierowanej informacji — tym bardziej, że w wyniku szybkiego tempa rozwoju gospodarki, gwałtownie wzrasta skala inwestycji infrastruktury technicznej na terenach zurbanizowanych.

*

We wspomnianym wyżej artykule na temat SSIM opisano dość szczegółowo podstawowe założenia podsystemu KUBATURA, którego zasięg tematyczny obejmuje planowanie i realizację wszystkich obiektów kubaturowych (w tym także i tych, które stanowią kubaturowe składniki infrastruktury technicznej). W podsystemie tym wyróżnia się moduły: PLAN, KONTROLA, OCENA. Podobną strukturę przewiduje się dla podsystemu INFRASTRUKTURA, choć właśnie jego specyfika skłania do dokładniejszego omówienia jego założeń i roli w całości SSIM.

Szczególną wagę w trakcie formułowania założeń podsystemu INFRASTRUKTURA zwracano na stworzenie bazy danych potrzebnych do merytorycznej analizy decyzji lokalizacyjnej (tzw. informacje o terenie oraz decyzje o czasie i warunkach realizacji inwestycji), które mogłyby szeroko korzystać ze standardowych nieomal już procedur optymalizacyjnych (programowanie liniowe, metody sieciowe itp.) gdyby nie chroniczny brak odpowiednich danych. Z praktycznych zastosowań optymalizacji w planowaniu urbanistycznym wynika, że stosunek nakładów pracy na zebranie danych do czasu pracy komputera kształtuje się jak 1000:1. Chcąc więc wykorzystać możliwości optymalizacji, która może przynieść milionowe oszczędności, staje się konieczne utworzenie dobrze zorganizowanego systemu informatycznego, który umożliwiałby łatwy dostęp do rzetelnych i aktualnych danych, przede wszystkim dla potrzeb planowania inwestycji.

Dlatego też w podsystemie INFRASTRUKTURA, posiadającym także moduły PLAN, KONTROLA i OCENA jedynie moduł KONTROLA może mieć podobną funkcję i strukturę jak analogiczny moduł w podsystemie KUBATURA.

Wynika to z następujących powodów:
— jedynie w tym module, którego podstawową funkcją jest bieżąca kontrola procesu realizacji inwestycji, można zakładać daleko posunięte podobieństwo struktury i funkcjonowania obu modułów, gdyż sposoby finansowania, przygotowania i sprawozdawczości z zakresu inwestycji kubaturowych są identyczne lub bardzo podobne do inwestycji z zakresu infrastruktury
— planowanie inwestycji kubaturowych oraz ocena skutków ich realizacji są już odmienne, gdyż:

- inwestycje kubaturowe wynikają z bezpośrednio przewidywanego rozwoju układów osadniczych i są w stosunku do infrastrukturalnych pierwotne
- inwestycje kubaturowe dają się na ogół sumować dla określonego obszaru i okresu, wewnątrz których istnieje znaczna wzajemna substytucja poszczególnych zadań. Rozumieć to należy następująco: cele rozwoju społecznego miasta (obszaru osadniczego) wyrażają się uzyskaniem pożądanego średniego standardu w określonym okresie. Jest więc dla tak wyrażonego celu w zasadzie obojętne, które budynki będą wcześniej, a które później realizowane, byleby przewidywany przyrost substancji mieszkaniowej i usługowej następował w pożądanej wielkości. Istotniejszą tu sprawą jest uzyskanie właściwej struktury przyrostu, niż ścisłe przestrzeganie kolejności realizacji zadań kubaturowych. Uwarunkowania dotyczące kolejności realizacji kubaturowych wynikać mogą więc przede wszystkim ze współzależności strukturalnych założonego standardu
- koszty kubatur nie są w zasadzie zróżnicowane przestrzennie (wyjąwszy koszt transportu materiałów budowlanych)
- inwestycje infrastruktury nie są wzajemnie substytucyjne w tym stopniu co kubaturowe, zaś sumowanie ich dla określonych obszarów lub okresów ma bardzo ograniczony sens
- infrastruktura techniczna przesądza optymalne — ze względu na koszty — rozmieszczenie w przestrzeni i w czasie elementów kubatury, choć sama bezpośrednio nie realizuje celów rozwoju społecznego
- koszty infrastruktury są przestrzennie zróżnicowane, a kolejność realizacji jej elementów jest zdeterminowana technicznie.

Mając powyższe na uwadze, staje się oczywiste, że jedynie moduł KONTROLA może być projektowany w obu podsystemach na identycznych zasadach, które w rezultacie powinny doprowadzić do ich dużego podobieństwa w zakresie zastosowanych programów, organizacji zbierania danych i przewidywanych informacji wyjściowych.
Jasne jest bowiem, że tryb kontroli i rozliczania realizacji zadań inwestycyjnych, ze względu na obowiązujące przepisy jest w tych przypadkach bardzo zbliżony. Pozostałe moduły różnią się między sobą dość znacznie ze względu na wspomniany odmienny w tym względzie charakter obu rodzajów inwestycji.

CEL I ZAKRES PODSYSTEMU INFRASTRUKTURA

Celem podsystemu jest obsługa informatyczna planowania, kontroli i oceny inwestycji infrastruktury technicznej związanej z rozbudową miasta, a świadczona na rzecz wszystkich instytucji, które biorą udział w procesie inwestycyjnym.

Zakres i główne funkcje podsystemu wynikają:

— z zakresu kompetencji jednostek prowadzących inwestycje infrastruktury miejskiej

— z charakteru technicznego i uwarunkowań inwestycji infrastruktury miejskiej

— z powiązań technicznych między inwestycjami infrastruktury i kubatury.

Zakres podsystemu wynikający z kompetencji jednostek prowadzących inwestycje infrastruktury obejmuje:

— systemy techniczne wodociągów wraz z urządzeniami służącymi do poboru, uzdatniania, przechowywania i przesyłu wody, które leżą w gestii miejskich (komunalnych) przedsiębiorstw wodociągów i kanalizacji. Wynika stąd, że poza systemem znajdują się inwestycje gospodarki wodnej państwa oraz inwestycje wodociągowe będące własnością pozakomunalną (np. zakłady przemysłowe, osoby prywatne, gospodarstwa rolne — państwowe lub spółdzielcze, które mogą prowadzić inwestycje w zakresie systemów wodociągowych wyłącznie na własny użytek

— systemy techniczne kanalizacji na podobnych zasadach co wodociągowe z tym, że wyłączenie z objęcia podsystemem informatycznym może nastąpić jedynie wtedy gdy:

- odbiornik ścieków służy tylko i wyłącznie jednostce eksploatującej wyodrębniony system kanalizacji
- tylko jednostka eksploatująca kanalizację korzysta (w jakikolwiek sposób) z danego odbiornika ścieków
- cały system techniczny (sieć, urządzenia i odbiorniki) leży w całości na terenie jednostki eksploatującej dany system kanalizacyjny.

Dotyczy to zarówno kanalizacji ogólnospławnej jak i rozdzielczej oraz specjalnych systemów (np. odprowadzenie wód kopalnianych lub gruntowych).

— systemy techniczne gazu na podobnych zasadach co systemy kanalizacji z wyłączeniem magistralnych rurociągów przesyłowych gazu ziemnego, które w tym przypadku należy traktować jako inwestycje przesyłowe produkcyjne, a nie inwestycje komunalne lub obsługujące wspólnie gospodarkę miejską i przemysł.

Dodatkowym elementem systemu technicznego, który powinien być objęty podsystemem INFRASTRUKTURA są źródła gazu (gazownie) o ile mają one obsługiwać gospodarkę komunalną, nawet jeżeli oprócz tej gospodarki obsługują inne zespoły

— systemy techniczne energii elektrycznej obejmują w zasadzie tylko sieć rozdzielczą z przynależnymi jej urządzeniami. Wynika to z faktu, że sieć energii elektrycznej w Polsce jest całkowicie zcentralizowana, a więc nawet niewielkie źródła czy też sieci energii, które mogłyby stanowić samodzielne miejskie systemy są włączone do sieci ogólnokrajowej

— Systemy techniczne energetyki cieplnej obejmują zarówno źródła jak i sieci cieplne przesyłowe i rozdzielcze o ile nie występuje podobne wyodrębnienie jak opisane w przypadku kanalizacji

— systemy techniczne łączności objęte podsystemem INFRASTRUKTURA to cała sieć miejscowa z pominięciem sieci i central międzymiastowych

— systemy techniczne komunikacji najbardziej rozbudowane i zróżnicowane.

Wewnętrzny podział tego systemu technicznego przebiega według zasady:

- sieć drogowa — komunalna
- sieć szynowa, z dalszym ewentualnym podziałem na tramwaj, kolej miejską lub metro
- urządzenia „źródłowe" komunikacji miejskiej — zajezdnie, tabor itp.

Techniczne uwarunkowania inwestycji infrastruktury miejskiej wymagają:

— opisu sieci i istniejących rezerw oraz ewentualnych możliwości rozbudowy

— opisu źródeł, istniejących rezerw i możliwości ich powiększenia.

Powiązania między inwestycjami kubaturowymi a infrastrukturalnymi dotyczą wszystkich rodzajów infrastruktury i wszystkich inwestycji kubaturowych. Poniesione lub planowane inwestycje infrastruktury decydują o możliwościach i efektywności lokalizacji inwestycji kubaturowych. Nadają one bowiem przestrzeni zurbanizowanej konkretną wartość, przejawiającą się w tzw. korzyściach zewnętrznych lokalizacji. Zagadnienia te zostały omówione szerzej w mojej pracy pt. „Optymalizacja lokalizacji na terenach zurbanizowanych", OBRI 1975.

Wydaje się, że wystarczy pamiętać, iż w zagadnieniach planowania infrastruktury technicznej miast powstają dwa zasadnicze problemy:

— poszukiwanie optymalnej lokalizacji dla zadanego programu inwestycyjnego (odpowiedź na pytanie: gdzie?)

— poszukiwanie optymalnego sposobu użytkowania określonych obszarów czy działek (odpowiedź na pytanie: co?). Problem trzeci (odpowiedź na pytanie: kiedy?) nie był dostatecznie eksponowany w literaturze. Istniały bowiem dotychczas wystarczające rezerwy terenowe dające z jednej strony możliwość manewru planistycznego, z drugiej zaś skala inwestycji infrastrukturalnych upoważniała do podporządkowania się w tym względzie planowaniu inwestycji kubaturowych, warunkowanych najczęściej przez możliwości ich realizacji, tak ze względu na wielkość planowanych nakładów inwestycyjnych ogółem, jak i możliwość wykonawstwa inwestycyjnego. Obecnie w większości przypadków wystąpi zjawisko warunkowania inwestycji kubaturowych przez inwestycje infrastruktury w dużej skali. Stąd problem kolejności i terminów realizacji zaczyna nabierać szczególnego znaczenia w polityce inwestycyjnej poszczególnych miast, determinowanej coraz silniej przez pilną potrzebę wydatnego podniesienia standardów (komunikacja, ciepłownictwo), przy jednoczesnym szybkim rozwoju przestrzennym.

ELEMENTY PODSYSTEMU INFRASTRUKTURA

Struktura podsystemu INFRASTRUKTURA jest w swoim ogólnym zarysie bardzo zbliżona do podsystemu KUBATURA, choć cele i zakresy modułów PLAN i OCENA są odmienne.

Moduł PLAN — INFRASTRUKTURA

Celem tego modułu jest obsługa informatyczna służb miejskich w zakresie planowania inwestycji infrastruktury technicznej.

W tym względzie należy wydzielić:

— jednostki funkcjonalne obsługujące merytoryczne przebiegi formalne planowania i zapewniające wewnętrzna zgodność planu, a także jego zgodność z obowiazujacymi przepisami i normami. Szczególnie należy tu zwrócić uwagę na liczne i skomplikowane uzgodnienia

— jednostki funkcjonalne obsługujace merytoryczne przebiegi planowania infrastruktury miejskiej, do których zaliczyć należy przede wszystkim:

- badanie efektywności inwestycji
- generowanie wariantów inwestycyjnych
- symulację systemów technicznych, szczególnie komunikacji masowej
- optymalizację rozmieszczenia inwestycji i ich etapowania tak ze względu na koszty i możliwości realizacyjne poszczególnych fragmentów systemów technicznych jak i na możliwości przyszłej dalszej rozbudowy

— moduł PLAN obsługiwać powinien różne horyzonty planowania:

- plan perspektywiczny obejmujący okres 15—20 lat zgodnie z horyzontem planu ogólnego
- plan etapowy (planowania przestrzennego), który będzie zawsze planem postulatywnym w stosunku do planu gospodarczego
- plan 5-letni inwestycyjny obejmujący te inwestycje, z planu etapowego, które znalazły pokrycie w limitach inwestycyjnych
- plan roczny — inwestycyjny będący wycinkiem planu 5-letniego.

Z powyższego wynika, że struktura wewnętrzna modułu PLAN — INFRASTRUKTURA będzie dość skomplikowana, gdyż musi uwzględniać:

— poszczególne branże infrastruktury
— poszczególne przebiegi formalne
— odpowiednią fazę przebiegów merytorycznych
— właściwy horyzont planistyczny.

Należy zwrócić uwagę na potrzebę odpowiedniej agregacji danych w poszczególnych horyzontach czasowych.

Należy również pamiętać o zapewnieniu możliwości przejścia podsystemu INFRASTRUKTURA na planowanie kroczące, wymagające ciągłego (lub okresowego lecz częstszego) powiększania szczegółowości danych przy przenoszeniu ich z poziomu planu o dłuższym horyzoncie do planu o horyzoncie krótszym. To zaś może wymagać innych, bardziej skomplikowanych procedur wykorzystujących odpowiednie miary nieokreśloności i przejścia z planowania deterministycznego na planowanie probabilistyczne.

Teoretyczne przesłanki takiego planowania są znane (por.: Jay W. Farrester — M.I.T., PRESS 1969 oraz J. Z. Majminas — Procesy płanirowanija w ekonomikie, Izdatielstwo Ekonomika, Moskwa 1971 r.). Problem leży więc w dostosowaniu teorii do naszych potrzeb i możliwości — przede wszystkim organizacyjnych oraz sprzętowych.

Szczególną uwagę warto poświęcić modułowi PLAN — INFRASTRUKTURA z tego względu, że może stać się on źródłem uzyskiwania konkretnych korzyści ekonomicznych. Oszczędności nakładów inwestycyjnych uzyskiwane w procesie poprawiania lub optymalizacji decyzji z tego zakresu sięgać mogą znacznych sum, które mogłyby z łatwością pokryć koszty budowy systemu informatycznego. Problem zaś w tym, że uzyskanie tych korzyści nie może dotyczyć wyłącznie „oszczędności papierowych", wymaga więc nie tylko umiejętności podejmowania optymalnych decyzji planistycznych, ale również ich konsekwentnej realizacji, czego ani sam moduł PLAN ani informatyka nie zapewni. Stąd wynika konieczność kompleksowego, choć sukcesywnego, tworzenia podsystemu INFRASTRUKTURA.

Użytkownikami modułu powinny być:
— miejskie komisje planowania
— pracownie urbanistyczne
— miejskie służby inwestycyjne.

Pozostali użytkownicy, tzn. jednostki eksploatujące poszczególne systemy techniczne powinny korzystać z modułu PLAN — INFRASTRUKTURA za pośrednictwem wyżej wymienionych instytucji.

Moduł KONTROLA — INFRASTRUKTURA

Moduł ten nie wymaga specjalnego opisu. Odwołując się do cytowanego artykułu o SSIM — KUBATURA, wystarczy podać podstawowe założenia jego współpracy z pozostałymi modułami podsystemu INFRASTRUKTURA. Są one następujące: na wejściu modułu KONTROLA powinny się pojawiać wyłącznie te przedsięwzięcia i zadania, które zostały ostatecznie ustalone w jednostce „plan roczny" modułu PLAN.

Struktura informacji identyczna jak w podsystemie KUBATURA z tą tylko różnicą, że systemy techniczne infrastruktury muszą być uporządkowane hierarchicznie. Stąd potrzeba notowania zawsze miejsca, które zajmuje określony obiekt w systemie technicznym. Najwygodniej jest robić to w postaci tablicy krzyżowej podobnej jak dla zapisu macierzowego sieci powiązań.

W podobny sposób należy też powiązać kompleksy inwestycji kubaturowych z odpowiednimi elementami infrastruktury. Należy liczyć się z dość znacznymi wymiarami takich tablic (rzędu kilka tysięcy na kilka tysięcy). Dopiero po sprawdzeniu stopnia zapełnienia tabeli można próbować stosowania oszczędniejszych form zapisu — np. przechodząc na wielowymiarową przestrzeń, której liczba wymiarów odpowiada liczbie branż infrastruktury zaś obiektem identyfikowanym w ten sposób jest każdy obiekt kubaturowy. Można będzie każdemu zestawowi inwestycji kubaturowych przypisać niezbędne dla jego realizacji i użytkowania elementy infrastruktury i w ten sposób znajdować podstawę do ocen funkcjonalnych i ekonomicznych.

Moduł OCENA — INFRASTRUKTURA

Celem modułu jest bieżąca i okresowa ocena:
a) realizacji planu
b) prawidłowości planu jako wzorca
c) stopnia realizacji celów nadrzędnych planu
d) uzyskiwanych wskaźników techniczno-ekonomicznych.

Ad. a) Ocena realizacji planu polega na bieżącym notowaniu zgodności danych rzeczywistych z wzorcem, który stanowi w tym przypadku plan. Ocena dotyczy więc przede wszystkim tych wielkości, które stanowią podstawową treść planu. Będzie ona zatem różna dla określonych szczebli planowania. Trudność tu występująca polega na zmienności planów, a co za tym idzie, konieczności elastycznego projektowania modułu OCENA z uwzględnieniem wysokiego stopnia parametryzacji.

Zasadą, która musi być bezwzględnie zachowana jest uzyskiwanie danych dla tego modułu wyłącznie poprzez moduły PLAN i KONTROLA.

Inne podejście niweczyłoby bowiem sens całego systemu SSIM. Nie oznacza to jednak, by poszczególne moduły czy jednostki informatyczne nie mogły funkcjonować autonomicznie. Jednakże cały system powinien czerpać dane ze źródeł modułu PLAN do tworzenia wzorca oraz ze źródeł modułu KONTROLA do tworzenia obrazu stanu rzeczywistego.

Z powyższego założenia wynika, że obraz stanu rzeczywistego musi być tworzony w drodze agregacji obrazów niższych szczebli lub krótszych okresów. Osobną sprawą, będącą wyłączną procedurą modułu OCENA jest ustalenie kryteriów oceny pozytywnej lub negatywnej i sposób generowania alarmów. Przyjmuje się, że w ocenie realizacji planu każde odchylenie od wzorca jest notowane, nie każde jednak musi wywoływać alarm. Zakłada się zasadę dopuszczalnej kompensacji odchyleń od wzorca bądź to w poszczególnych okresach lub obszarach, bądź to dla poszczególnych podmiotów gospodarczych. Alarm na szczebel wyższy generuje się wyłącznie wtedy, gdy suma elementów szczebla niższego daje odchylenia większe od dopuszczalnych.

Ad. b) Ocena prawidłowości i jakości planu występuje w zasadzie wtedy, gdy ocena realizacji wykazuje odchylenia (in plus lub in minus) od wzorca. Zachodzi wówczas domniemanie, że plan — wzorzec został ustalony nie dość precyzyjnie. Do takiej oceny konieczne jest uzyskanie danych o wszystkich notowanych odchyleniach — niezależnie od tego czy wywołały one alarmy czy też nie.
Ocenę taką należy prowadzić w stosunku do wszystkich wersji planu. Stąd potrzeba ich przechowywania — nawet jeżeli dana wersja planu stała się nieaktualna wskutek wprowadzonych zmian.

Archiwalne wersje planu można przechowywać — bądź to w module PLAN bądź w module OCENA.

Wydaje się, że przechowywanie ich w module OCENA będzie bardziej prawidłowe, jako że moduł ten z założenia będzie musiał gromadzić dość obszerne zbiory celem uzyskiwania wskaźników z różnych układów.

Ad. c) Ocena realizacji celów nadrzędnych planu zakłada możliwość uzyskiwania danych dotyczących skutków realizacji planów, lecz w nieco innej płaszczyźnie niż proste stwierdzenie wykonania zadań ilościowych. Podobnie jak poprzednio cele nadrzędne mogą być formułowane dość różnie — należy zatem liczyć się tutaj również z koniecznością daleko posuniętej parametryzacji. Najogólniej można zakładać, że cele nadrzędne planu będą dotyczyć długookresowych standardów wyrażających się np.:

— średnią wielkością mieszkań
— średnim zagęszczeniem izb
— zespołem standardów osiedlowych w zakresie usług.

W zakresie infrastruktury technicznej należy liczyć się z następującymi standardami:

— zdolność produkcjna (przesyłowa) całej sieci danego systemu technicznego
— zróżnicowanie dopuszczalnych poziomów zapotrzebowania w czasie i przestrzeni
— średnia zdolność produkcyjna (przesyłowa) w przeliczeniu na jednostkę odniesienia (jednostka powierzchni lub jednostka „użytkownika" — np. na 1 mieszkańca).

Dotyczy to wszystkich systemów technicznych infrastruktury — z tym, że w zakresie komunikacji osobowej staje się to zagadnienie zbyt skomplikowane — aby można było omówić je dostatecznie w niniejszym opracowaniu. Szczegółowe zasady oceny komunikacji miejskiej można znaleźć w pracy IOMB wykonanej pod moim kierownictwem pt. „Badania nad rozmieszczeniem budownictwa mieszkaniowego w Krakowie na tle dojazdów do pracy" wydanej w 1970 r.

Podane tam zasady oceny potencjałów oraz efektywności ekonomicznej powinny być zastosowane do wszystkich branż miejskich, oczywiście z odpowiednimi modyfikacjami.

Ad. d) Moduł OCENA powinien dysponować własną biblioteką danych pozwalających na szybkie uzyskiwanie wskaźników techniczno-ekonomicznych dla potrzeb oceny stanu czy realizacji, jak też dla konstrukcji planu czy jego aktualizacji.

Dane powinny być klasyfikowane w oparciu o zasadę identyfikacji organizacyjnej inwestora, technologicznej i adresowej obiektu oraz czasu jego powstania i wykonawcy. Dane pierwotne powinny być przechowywane w taki sposób aby uzyskiwanie potrzebnych wskaźników nie natrafiało na trudności.

MOŻLIWOŚCI REALIZACJI

Jak z powyższego widać najbardziej rozbudowanym modułem będzie KONTROLA wraz ze źródłami danych oraz obsługą sprawozdawczości GUS i NBP. Jest to zresztą moduł warunkujący prawidłową pracę pozostałych modułów i w największym stopniu autonomiczny. Dlatego też powinien być wdrażany jako pierwszy, łącznie z niektórymi elementami PLANU i OCENY.

Największe korzyści gospodarcze można uzyskać dzięki modułowi PLAN, którego funkcjonowanie uwarunkowane jest przede wszystkim przez KONTROLĘ i OCENĘ.
Trzeba przy tym zwrócić uwagę, że powyżej opisany system informatyczny jest systemem wyodrębnionym według procesu, który obserwuje i którego sterowaniu ma służyć. Nie jest on natomiast pomyślany jako system informatyczny poszczególnej jednostki gospodarczej czy administracyjnej.

Dlatego nie można zakładać, że występować tu będzie jeden użytkownik. Wprost przeciwnie należy się liczyć z wieloma użytkownikami, którymi przede wszystkim powinny być:

— miejskie komisje planowania, służby inwestycyjne i pracownie urbanistyczne w zakresie modułu PLAN
— służby inwestycyjne w zakresie modułu KONTROLA
— miejskie komisje planowania i pracownie urbanistyczne w zakresie modułu OCENA.

Stąd wydaje się, że podsystem powinien być traktowany jako system informatyczny wojewody służący sterowaniu rozwojem infrastruktury technicznej miast, udostępniający wszystkim zainteresowanym gromadzone w nim informacje.

Wydaje się, że w układach terytorialnych tylko systemy informatyczne służące wielu użytkownikom mogą dać właściwe efekty. Jakby bowiem nie traktować władzy i administracji terenowej, to zawsze podstawową jej funkcją będzie przede wszystkim koordynacja działania różnych branżowo (i resortowo) jednostek gospodarczych. Stąd też i systemy informatyczne obsługujące władzę i administrację terenową powinny służyć wszystkim podmiotom gospodarczym z danego obszaru.

PRZEGLĄD WYDAWNICTW

Bibliografia wydawnictw polskich z dziedziny informatyki

Obliczenia hybrydowe — BEKEY G. A., KARPLUS W. J. Tłum. wyd. ang. z roku 1968. WNT, Warszawa 1976, s.. 475, cena zł 90.—

Cz. 1. Teoria i środki realizacji: wprowadzenie w hybrydowe techniki obliczeniowe, części składowe hybrydowego systemu liczącego, numeryczne rozwiązanie równań różniczkowych zwyczajnych, metoda przekształceń w analizie hybrydowych systemów liczących, analiza błędów w hybrydowych systemach liczących, kompletne hybrydowe systemy liczące, oprogramowanie maszyn hybrydowych. Cz. 2: Zastosowanie maszyn hybrydowych: rozwiązywanie zagadnień polowych za pomocą maszyn hybrydowych, optymalizacja parametryczna, zagadnienia sterowania optymalnego, procesy przypadkowe, obliczenia hybrydowe w modelowaniu lotu, układy współpracy człowieka z maszyną, układy biologiczne, rozwiązanie równań całkowych.
Książka przeznaczona jest dla inżynierów automatyków oraz dla studentów wyższych szkół technicznych. Wymagane jest ogólne przygotowanie w zakresie analogowych i cyfrowych metod obliczeniowych.

Systemy przesyłania informacji cyfrowych — SEIDLER J. Wyd. 2 zmien. WNT, Warszawa 1976, s. 618, cena zł 117.—

Opis systemów przesyłania informacji cyfrowych. Optymalizacja systemów przesyłania informacji cyfrowych. Własności ogólne kodów liniowych. Kody cykliczne. Kody rekurencyjne. Własności ogólne systemów z kanałami ciągłymi. Systemy optymalne z kanałami ciągłymi. Systemy ze sprzężeniem zwrotnym decyzyjnym. Systemy ze sprzężeniem zwrotnym informacyjnym. Systemy wielodostępne. Dodatki: 1. Podstawowe pojęcia teorii przestrzeni liniowych. 2. Obliczanie prawdopodobieństw stanów buforu z opróżnianiem ciągłym. 3. Tablica kodów cyklicznych dopuszczających niezależne zespoły ocen. W drugim wydaniu zmieniono układ rozdziału 2 oraz wprowadzono drobne poprawki do innych rozdziałów. Pominięto szczegóły wyprowadzeń niektórych wzorów, podano informacje w jakich publikacjach wyprowadzenia te można znaleźć. Usunięto również pewne fragmenty pierwszego wydania. Szerzej opracowano zagadnienia związane z łącznością między komputerami cyfrowymi. Książka przeznaczona jest dla inżynierów pracujących w dziedzinie telekomunikacji i informatyki.

Zastosowanie programu MPS na IBM 370 przy obliczaniu zdolności produkcyjnych odlewni — SKARŻYŃSKA B. Wyd. Stowarzyszenia Inżynierów i Techników Mechaników Polskich, Warszawa 1976, s. 4. Studium zaoczne III stopnia.

Opis systemu MPS REZERWY. Dane wejściowe. Algorytmy. Graniczne wymiary zadań. Model odlewni z wariantową technologią. Praca zakłada, że czytelnik zna zasady budowy optymalizacyjnych modeli liniowych. Znajomość SIMPLEX ZAM, chociaż nie wymagana, może być pomocna przy zrozumieniu działania systemu MPS REZERWY. Materiały przeznaczone są na studium doskonalenia kandydatów na rzeczoznawców SIMP ds. wykorzystywania rezerw produkcyjnych w przemyśle maszynowym.

Transmisja danych. Tłum. wyd. ang. Wyd. Ośrodka Badawczo-Rozwojowego Informatyki INFORNA, Warszawa 1976 r., s. 49, cena zł 92. Europejski Program Badawczy Diebolda. Zeszyt 75 (E 148 M).

Wprowadzenie. Branża transmisji danych: struktura i problemy reglamentacji: struktura branży, zagadnienia reglamentacji i polityki państwa — skutki na przyszłość. Przegląd techniki: elementy składowe, urządzenia i nośniki transmisji. Podsumowanie: wnioski i skutki. Materiały przeznaczone są głównie dla kadry kierowniczej przedsiębiorstw i kierowników planowania działalności informatycznej.

Architektura logiczna EMC JS — KAMBURELIS T. Wyd. Ośrodka Badawczo-Rozwojowego Informatyki INFORNA, Warszawa 1976, s. 237, cena zł 53. Problemy informatyki.

Podstawowe cechy architektury logicznej. Organizacja kanałów wejścia-wyjścia. Charakterystyka jednostek centralnych JS EMC (modele: JS-1010, JS-1021, JS-1030, JS-1040, JS-1050). Model R-30U. Pamięć buforowa. Pamięć wirtualna. Urządzenia zewnętrzne. Zgodność programowa i emulacja EMC ODRA 1300. Załączniki: lista instrukcji JS EMC, metoda obliczania wydajności procesorów JS EMC (interpretacja mieszanki GIBSON I). Materiały przeznaczone są dla projektantów i programistów oraz dla elektroników zajmujących się eksploatacją komputerów Jednolitego Systemu.

Oprac. A.K.

III Krajowy Przegląd Zastosowań Techniki Komputerowej w Przemyśle Maszynowym. Poznań 19 – 21. X. 1976 r.

Dziewiętnastego października 1976 roku w poznańskim Pałacu Kultury została otwarta konferencja naukowo-techniczna na temat zastosowań maszyn matematycznych w przemyśle maszynowym.

Konferencja zorganizowana była pod patronatem ministra Maszyn Ciężkich i Budowlanych inż. Franciszka Adamkiewicza.

Trud organizacyjny wzięło na siebie Stowarzyszenie Inżynierów i Techników Mechaników Polskich Oddział w Poznaniu oraz Stowarzyszenie Księgowych Polskich i Polskie Towarzystwo Cybernetyczne. W obradach obok delegatów z Polski wzięli także udział przedstawiciele organizacji technicznych z Węgier, NRD i ZSRR.

Zorganizowany już po raz trzeci (po raz pierwszy w 1966 r.) przegląd zastosowań komputerów w przemyśle maszynowym miał umożliwić uczestnikom, a za ich pośrednictwem także zainteresowanym inżynierom i technikom oraz informatykom w kraju, wymianę doświadczeń zdobytych w ciągu pięciu lat, jakie upłynęły od poprzedniej konferencji. Wnioski z dyskusji powinny wytyczyć kierunki rozwoju i zastosowań techniki komputerowej w przemyśle maszynowym. Posiedzenia sesji naukowo-technicznych poprzedzone były serią następujących referatów wygłoszonych podczas sesji plenarnej:

- prof. dr inż. Janusz Dietrych — Kryteria wspomagania komputerowego w działaniach technicznych
- mgr inż. Waldemar Stroiński — Organizacyjne przygotowanie przedsiębiorstwa do stosowania EPD w procesie zarządzania
- dr inż. Władysław Rok, mgr inż. Hubert Wyszko — Informacyjny system sterowania jakością wyrobów w przedsiębiorstwie budowy maszyn
- dr inż. Juliusz Błoński, dr inż. Krzysztof J. Jesionek, dr inż. Andrzej Sandecki, prof. dr hab. inż. Ryszard Wyszyński, mgr inż. Jan Zabdyr — Zastosowanie ETO do wyznaczania charakterystyki wentylatora
- inż. Jerzy Bielówka — Przegląd prac dla przemysłu maszynowego wykonywanych w ZETO Poznań
- dr inż. Zbigniew Dobrzyński — Organizacyjne przygotowanie użytkowników w biurze projektowym
- prof. dr inż. Seweryn Chajtman — Logiczne podstawy projektowania systemów informatyki.

Podobnie jak na poprzednich konferencjach obrady odbywały się w czterech sekcjach:
— przemysłowych zastosowań techniki komputerowej
— teorii systemów informatycznych
— organizacyjnego przygotowania użytkowników
— metodyki projektowania i oprogramowywania systemów informacyjnych.

W sekcji przemysłowych zastosowań techniki komputerowej obradowano nad zastosowaniami systemów EPD oraz nad problematyką obliczeń naukowo-technicznych i projektowanych.

W sekcji tej wygłoszono następujące referaty:

- dr inż. Krzysztof J. Jesionek — Wyznaczenie parametrów przepływowych w przekroju kontrolnym kanału zakrzywionego
- dr inż. Jerzy Sąsiadek — Badanie dynamiki urządzeń na przykładzie badania właściwości dynamicznych turbiny parowej
- mgr inż. Jan Jędryszek — Obliczenie charakterystyki hałasu od współpracy wirnika i kierownicy w maszynach osiowych
- mgr Wojciech Moczulski — Zastosowanie komputera dla analizy sygnałów w badaniach konstrukcyjnych maszyn
- mgr Adam Jerzembowski — Elektroniczne automaty obrachunkowe SOEMTRON i przykłady ich zastosowania
- dr Feliks Bator — Ogólna charakterystyka minikomputerowego systemu centralnej rejestracji i przetwarzania danych dla obrabiarek sterowanych numerycznie
- mgr inż. Jacek Kaliński — System minikomputerowy w biurze projektowym. Doświadczenia eksploatacyjne i wnioski
- dr inż. Henryk Kozłowiecki, dr inż. Janusz Przastek, mgr inż. Zdzisław Bogucki — System obliczenia parametrów pracy łożysk silników spalinowych
- dr inż. Stanisław Ziętarski, mgr inż. Jerzy Pytkowski — Komputerowa metoda doboru optymalnych parametrów sterowania
- dr inż. Andrzej Komorowski, mgr inż. Krzysztof Rumiński — Rozwój wspomaganych komputerem metod projektowania konstrukcji i technologii w przemyśle maszynowym
- mgr inż. Zdzisław Bogucki — Zarys komputerowego banku danych BS AGROMASZ.

W sekcji teorii systemów informatycznych obradowano nad: ogólną teorią systemów informatycznych, teorią optymalizacji, systemami wielowymiarowymi i wielopoziomowymi, wielkimi systemami informatycznymi, systemami adaptacyjnymi i samouczącymi się, a także nad zagadnieniami stochastycznymi i teorią algorytmów, matematycznym podejściem do projektowania, językami formalnymi i algorytmicznymi, metodami numerycznymi i teoretycznymi aspektami zagadnień wyszukiwania informacji.

W sekcji tej przedstawiono następujące referaty:

- mgr inż. Jerzy Mocała — Dekompozycja zbioru parametrów opisujących złożone obiekty techniczne
- Krzysztof Staniszewski — Wybrane problemy zarządzania zasobami w systemach wielodostępnych ze wspólną bazą danych
- mgr Jacek Wiłaszek — System automatycznej konstrukcji translatorów
- mgr Adolf Sołtysek — Alfanumeryczny język zapisu konstrukcji w sy-

FOTO 1. Za stołem prezydialnym od lewej: dr inż. Władysław Rok z Politechniki Poznańskiej, prof. dr inż. Janusz Dietrych z Politechniki Śląskiej — Przewodniczący Prezydium, mgr inż. Witold Kawecki oraz dyrektor ZETO w Poznaniu mgr Hieronim Postaremczak

FOTO 2. Prof. dr inż. Janusz Dietrych wygłasza pierwszy referat na temat: Kryteria wspomagania komputerowego w działach technicznych

stemie automatycznego projektowania technologii

- mgr inż. Andrzej Wachowiak — Transponowanie cech konstrukcyjno-technologicznych rysunku technicznego dla potrzeb przygotowania technologicznego produkcji
- dr inż. Sławomir Kotra — Formalizacja zadania optymalnego rozmieszczenia kontroli technicznych i poprawkowych sieci procesu technologicznego
- dr inż. Ryszard Pawlik — Konwersacyjny program optymalnego projektowania urządzeń technicznych
- inż. Jerzy Sukiennik — Czynniki decydujące o modelu informatyki w przedsiębiorstwie przemysłowym.

W sekcji organizacyjnego przygotowania użytkowników przedstawiono koncepcję docelowej struktury organizacyjnej przy zastosowaniu komputera w zarządzaniu przedsiębiorstwem, kombinatem lub branżą; zaprezentowano projekty przejścia z obecnego stanu organizacyjnego do stanu docelowego oraz reorganizacji przedsiębiorstwa dla wprowadzenia techniki komputerowej.

W sekcji tej wygłoszono następujące odczyty:

- mgr Jan Nowicki — Organizacja prac projektowo-programowych przy zastosowaniu tablic decyzyjnych
- mgr inż. Stanisław Kowalski — Przygotowanie baz danych indeksowych i normatywnych dla systemów informatycznych
- mgr inż. Stanisław Bekisz, mgr inż. Zbigniew Zajączkowski — Problemy eksploatacji i konstrukcji systemów informatycznych dla potrzeb ośrodków badawczo-rozwojowych
- dr Ignacy Dziedziczak — Modele organizacji księgowości w systemie informatycznym
- mgr inż. Włodzimir Karpiński — Metodyka optymalizacji doboru i podziału asortymentowego planu produkcji branży przemysłowej
- Krzysztof Staniszewski — Metoda optymalizacji programów wynikowych.

Sekcja metodyki projektowania i oprogramowywania systemów informacyjnych zajmowała się zagadnieniami: Stosowanie języków algorytmicznych, projektów wstępnych systemów EPD, projektów technicznych i oprogramowania systemów EPD oraz oprogramowania dla potrzeb obliczeń naukowo-technicznych i projektowych.

W tej sekcji wygłoszono następujące referaty:

- dr inż. Janusz Śliwieński, mgr inż. Bernard Buśko — Kontrola wiarygodności danych w branżowych systemach informatycznych
- dr inż. Andrzej Sokołowski — Ochrona zbiorów informacji i jej skuteczność w systemach EPD
- dr inż. Stanisława Bońkowicz-Sittauer — Dokumentacja programów użytkowych
- mgr Mirosława Malinowska, mgr Jakub Malinowski — Przygotowanie procesu przetwarzania w ośrodku obliczeniowym
- mgr Stanisław Baumann, mgr Alina Raciszewska — Wprowadzenie do problematyki komputerowych systemów zarządzania realizacją prac projektowo-programowych
- mgr inż. Andrzej Krowiak — Koncepcja wielopoziomowego hierarchicznego systemu komputerowego do zarządzania operatywnego produkcją w FSM
- mgr Zenon Mirski, mgr Ryszard Waltman, mgr Piotr Wilczyński — Ewidencja i kontrola pracy maszyn i urządzeń oraz zużycia części zamiennych w pokrywach maszyn górniczych
- mgr Barbara Herman, mgr inż. Jan Kudzbalski — Podsystem gospodarki częściami zamiennymi urządzeń produkcyjnych
- dr inż. Zdzisław Nowak — Zastosowanie techniki komputerowej do projektowania sieci elektroenergetycznych zakładów przemysłowych
- dr inż. Zdzisław Kabza, mgr inż. Bolesław Dobrowolski — Automatyzacja projektowania przepływomierzy zwężkowych.

Bardzo cenną i pożyteczną imprezą było zorganizowanie tak zwanych **„dyskusji okrągłego stołu"**, w których spotykali się bezpośrednio użytkownicy z producentami sprzętu, między innymi przedstawiciele Zjednoczenia Informatyki i Zjednoczenia MERA.

Dyskusje te odbywały się w trzech grupach: sprzętu komputerowego, oprogramowania oraz kadr informatyki. W toku wymiany poglądów wyłoniła się lista problemów wymagających pilnego załatwienia.

Między innymi w dyskusjach „okrągłego stołu" — (patrz sprawozdanie zamieszczone poniżej) — poruszana była sprawa poprawy stanu przekazywania informacji. Przedstawiciel Zjednoczenia Informatyki postulował zintensyfikowanie rozpowszechniania informacji przez wydawnictwa OBRI, INTE i inne. Niestety zapomniano (jak wynika ze sprawozdania) o istnieniu czasopisma INFORMATYKA, o postulatach sformułowanych w czerwcu 1976 r. na naradzie ZOINTE w Krakowie, o wnioskach ustalonych we wrześniu ubiegłego roku na konferencji INFOGRYF. Przypominamy, że redakcja czasopisma zaoferowała wówczas udostępnienie dwóch kolumn drukowanych w trybie ekspresowym.

Prace organizacyjne III Krajowego Przeglądu Zastosowań Techniki Komputerowej w Przemyśle Maszynowym rozpoczęto w lutym 1976 roku. Część materiałów konferencyjnych (tom drugi, części 1—3) uczestnicy otrzymali przed rozpoczęciem obrad, pozostałe części (tom pierwszy i trzeci oraz część czwarta tomu drugiego) zostaną rozesłane po zakończeniu konferencji.

Przed kolejnym czwartym Przeglądem Zastosowań Techniki Komputerowej w Przemyśle Maszynowym planuje się zorganizowanie konkursu na najlepsze rozwiązanie systemowe i najlepsze wdrożenie systemu informatycznego. Protektorat nad tym atrakcyjnym konkursem przyjął minister PMCiB inż. F. Adamkiewicz.

Ustalono, że wyniki tego konkursu zostaną ogłoszone w czasie IV konferencji.

Ilustracją przebiegu obrad w sekcjach oraz w dyskusjach okrągłego stołu jest zamieszczona niżej relacja.

ANDRZEJ KLIMEK
(tekst i zdjęcia)

Uwagi i spostrzeżenia uczestnika konferencji w Poznaniu

W obradach sekcji metodyki projektowania systemów informatycznych wygłoszono 11 referatów.

Duże zainteresowanie i żywą dyskusję wzbudził m.in. referat dr. inż. Stanisławy Bońkowicz-Sittauer pt. „Dokumentacja programów użytkowych" wskazujący, że zagadnienia dokumentacji zarówno programów, jak też pozostałej dokumentacji systemów informatycznych wymagają dalszych prac i ustaleń normatywnych.

Tematyka ta w szerszym ujęciu była również omawiana w drugim dniu obrad, podczas dyskusji „okrągłego stołu" (grupy: „sprzęt", oprogramowanie", „kadry i szkolenie").

W dyskusji grupy „oprogramowanie" prowadzonej przez dra inż. Zbigniewa Dobrzyńskiego, wziął udział inż. Ryszard Terebus, zastępca dyrektora Zjednoczenia Informatyki. W dyskusji, która rozpoczęła się od przeglądu stanu oprogramowania w Polsce stwierdzono, m.in., że w dalszym ciągu brak jest pełnej informacji o istniejących programach i systemach.

Dyr. Ryszard Terebus oprócz licznych odpowiedzi na zadawane pytania, przedstawił obszerną informację o zadaniach Zjednoczenia oraz o niektórych przedsiębiorstwach z zakresu informatyki w Polsce. Omówione zostały takie problemy jak: konieczność synchronizacji projektowania nowych systemów w różnych ośrodkach, w tym odstępowanie (odsprzedaż) istniejących, typowych systemów EPD, szczególnie z zakresu gospodarki materiałowej, gdzie występuje największe dublowanie rozwiązań, podziału produkcji i oprogramowania komputerów Jednolitego Systemu pomiędzy kraje członkowskie (Fundusz Oprogramowania Jednolitego Systemu). Przy poruszanym zagadnieniu wyjazdów za granicę, rozwinęła się dyskusja na temat poprawy stanu przekazywania informacji zdobytych doświadczeń przez osoby wyjeżdżające i rozpowszechniania tych informacji.

Dla porównania podano, że EPD DIEBOLDA rocznie publikuje ok. 9 tys. wydawnictw, natomiast w Polsce ukazuje się zaledwie ok. 300.

Jak bumerang powracała w dyskusji sprawa dokumentacji systemów EPD zwłaszcza stopnia jej szczegółowości, głównie w celu umożliwienia modyfikacji lub rozbudowy oprogramowania przez użytkowników. Nawet w tak wyspecjalizowanym gronie uczestników dyskusji zdania na temat były podzielone.

Ożywiona dyskusja, która przeciągnęła się w czasie, przyniosła jej uczestnikom spory zasób informacji z zakresu szeroko pojętej dziedziny oprogramowania. Sprecyzowanym wynikiem tej dyskusji były przedstawione wnioski, m.in. konieczność prowadzenia prac nad pakietem programów zastosowań gospodarczych dla JS EMC, ujednolicenie zasad obrotu oprogramowaniem pomiędzy ośrodkami na końcowych plenarnych obradach konferencji.

Ostatni, trzeci dzień poświęcony był obradom plenarnym, podsumowaniu i zakończeniu konferencji. Jednym z bardziej interesujących wystąpień był referat prof. dr inż. Seweryna Chajtmana pt. „Logiczne podstawy projektowania systemów informatyki", w którym po ogólnym, metodologicznym ujęciu pojęć systemu i informacji referent przedstawił tezę, że obecnie komputer „przystawiamy" do istniejących już struktur organizacyjno-funkcjonalnych, co daje tylko część możliwości wykorzystania komputera. Teza ta w dalszym ciągu nic nie straciła na swej aktualności.

W dyskusji nad referatami, poruszono takie sprawy jak: formalizacja dokumentacji programów oraz kto jest informatykiem (np. według danych ICL — informatyk to w 80% fachowiec z określonej dziedziny, a w 20% specjalista z zakresu informatyki), zagadnienia wdrażania systemów informatycznych i przygotowania użytkowników zbyt często zdarzają się u nas przypadki, że komputery stoją w skrzyniach, ponieważ użytkownik nie jest przygotowany do ich zainstalowania.

Ostatnia część konferencji poświęcona była sprawozdaniom z obrad sekcji i ze spotkań „okrągłego stołu".

Jeden z postulatów zgłoszonych pod adresem organizatorów konferencji to powołanie na następnej (czwartej) konferencji nowej sekcji podstawowych problemów informatyki. Wnioski i postulaty z tych sprawozdań ujęto w projekcie Uchwały Konferencji. Materiały z trzeciego dnia obrad znajdą się w III tomie: Podsumowanie Konferencji, który zostanie rozesłany do uczestników.

Na zakończenie obrad głos zabrał sekretarz naukowy Komitetu Organizacyjnego mgr inż. Zdzisław Bogucki, który kończąc swoje wystąpienie, zaprosił zebranych na IV Krajowy Przegląd Zastosowań Techniki Komputerowej w Przemyśle Maszynowym do zamku w Kwidzynie, będącego w końcowej fazie odbudowy. W tym miejscu można dodać, że w czasie obrad wyświetlany był kolorowy film o odbudowie zamku i uroczystym otwarciu jego części, dający wyobrażenie o tym pięknym obiekcie, otoczonym starym parkiem. Zamek (ok. 400 pokoi) odbudowany w znacznej mierze siłami członków SNT NOT, znajdzie się w użytkowaniu Naczelnej Organizacji Technicznej.

Zachęcam do zapoznania się ze starannie wydanymi, obszernymi materiałami konferencji [1]), które oprócz przeglądu zastosowań komputerów w przemyśle maszynowym, przedstawiają stosunkowo szeroki zakres innych zagadnień związanych z teorią i praktyką informatyki.

Andrzej Sokołowski

---

[1]) Organizatorzy konferencji (adres: Ośrodek Doskonalenia Kadr SIMP Oddział w Poznaniu, al. Stalingradzka 5/9) posiadają jeszcze pewien zapas materiałów konferencyjnych.

X-lecie ośrodka zrodzonego z potrzeb i entuzjazmu

Na przełomie października i listopada minionego roku do kręgu jubilatów w Zjednoczeniu Informatyki dołączyło ZETO w Białymstoku. Fakt ten dał asumpt do zorganizowania jubileuszowego seminarium i skromnych obchodów. Redakcji zaś do zaprezentowania jubilata.

Zakład Elektronicznej Techniki Obliczeniowej w Białymstoku powstał właściwie z niczego. Pierwszym i przez długi czas jedynym pracownikiem był jego obecny dyrektor mgr inż. Witold Szyszło, zaś za lokal zakładowi służyło prywatne, dyrektorskie mieszkanie.

Nowo powstały zakład miał do rozwiązania wiele problemów. Brak było sprzętu, kadry, budynku a zapotrzebowanie na jego usługi było tylko potencjalne. Teren Ziemi Białostockiej — przedwojenna Polska „B" — urbanizował się i uprzemysławiał w szybszym tempie niż przeobrażenia w nadbudowie. W sytuacji — gdy do wprowadzenia do zarządzenia i organizacji zakładów przemysłowych elektronicznej techniki obliczeniowej — chętnych należało werbować, tolerancja dla ewentualnych potknięć ZETO była z góry zminimalizowana. Zatem dorabiać się własnej biografii i tradycji należało ostrożnie, zaś z drugiej strony możliwie szybko.

W pół roku od daty oficjalnego powstania ZETO, liczący już czterech pracowników otrzymał przydział lokalu. Przystąpiono do jego adaptacji i jednocześnie do szkolenia kadry. Szkolenie, co jest oczywiste w świetle naszkicowanej wyżej sytuacji, przebiegało poza Białymstokiem. Pierwsi pracownicy przechodzili je w Warszawie, Gdyni i Wrocławiu a także w NRD i ZSRR.

Na początku 1967 r. do eksploatacji weszły dwa zestawy MLA (SOEMTRON) a trzydziestokilkuosobowa załoga przystąpiła do ich zagospodarowania.

Pierwszym reflektantem na usługi ZETO był Zakład Energetyczny, a w ślad za nim do grona użytkowników przystąpiły Zakłady Przemysłu Bawełnianego w Fastach i Białostockie Zjednoczenie Budowlane. Na rzecz Zakładów Energetycznych rozliczono odbiorców energii elektrycznej i gazu, dla Zjednoczenia Budownictwa prowadzono obrót materiałowy oraz ewidencję zapasów magazynowych.

Staranna obsługa pierwszych klientów oraz szybko rozpowszechniająca się opinia o korzyściach płynących ze stosowania ETO, rychło zaczęły przysparzać ZETO dalszych kontrahentów.

W niespełna dwa lata od rozpoczęcia eksploatacji pierwszych systemów, z początkiem 1970 r., było już oczywiste, że zestawy MLA mimo 3-zmianowej pracy nie są w stanie podołać rosnącym zamówieniom. Zwłaszcza, że na skutek szkoleń prowadzonych siłami ZETO dla przedstawicieli kluczowych Zakładów regionu, idea ETO znajdowała coraz to nowych zwolenników.

Jednakże mimo rosnącego zapotrzebowania, mimo spiesznego przeszkalania kadry i przygotowywania systemów (na spodziewany komputer) w szeregu ośrodków w kraju, dopiero w 1974 roku miał on pojawić się w Białymstoku.

FOTO 1. 2 zestawy ODRY 1305 wypełniają dość szczelnie salę komputerową ZETO w Białymstoku. Miejsca na otrzymanego ostatnio RIADA już nie starczyło — trzeba było go ulokować gdzie indziej. Choć pod względem warunków lokalowych ZETO nie jest specjalnie upośledzone, dobrze, że zawczasu pomyślało o dalszej przyszłości i zabezpieczyło warunki pod budowę nowej siedziby. Realizacja inwestycji jeszcze trochę potrwa, a za parę lat Białystok wzbogaci się o efektowny budynek (wznoszony na wzór Centrum ZETO w Łodzi).

Oczywiście z perspektywy czasu można zastosować stare porzekadło, że „nie ma tego złego co by na dobre nie wyszło", bowiem istotnie tak długi okres wyczekiwania przyczynił się do doskonałego przygotowania gruntu pod instalację komputera. Wszelako należy sobie uświadomić jak trudny był to okres dla ośrodka: ile ważnych inicjatyw na rzecz użytkowników trzeba było wówczas zaniechać.

We wrześniu 1974 roku w zaadaptowanym na ten cel budynku przy ulicy Malmeda weszła do eksploatacji ODRA 1305.

W tym okresie ZETO nie zaniedbywało szans dalszej ekspansji na obszarze regionu i prowadziło starania nad powołaniem filialnych ośrodków w Łomży, Suwałkach i Ełku. Myślano także nad zabezpieczeniem dopływu kadry w świetle planów powiększania bazy sprzętu.

W tym celu uruchomiono we wrześniu 1974 r. pomaturalne dwuletnie studium programowania maszyn cyfrowych (w trybie stacjonarnym) i podobne lecz szkolące w zakresie przetwarzania danych studium zaoczne. Rozwiązywało to w zasadzie problem zapotrzebowania na średnią kadrę informatyczną; nadzieje na dopływ kadry projektantów i analityków wiązano z lokalną Politechniką.

Wracając do ODRY 1305, która pojawiła się w standardowej konfiguracji (pamięć operacyjna — 64 k, 6 jednostek pamięci taśmowej, 1 czytnik kart, 1 drukarka wierszowa) należy wspomnieć, że od pierwszych dni eksploatowana była przez dwie zmiany, zaś po upływie roku — na pełne trzy zmiany. Ku pełnej satysfakcji klientów z epoki MLA, sprawnie przeprogramowano systemy z SOEMTRONÓW i ODRY 1103 na komputer z prawdziwego zdarzenia. Niestety zaczęły się mnożyć awarie (jedna unieruchomiła ODRĘ na trzy tygodnie) powodujące znaczne kłopoty w obsłudze klientów.

Szczęściem odpowiednie władze zrozumiały trudną sytuację ZETO i bliźniacza ODRA 1305 wspomogła znakomicie park maszynowy, wchodząc do eksploatacji w marcu 1975 r. Aktualnie potencjał sprzętu został zwiększony przez zainstalowanie komputera JS R-20 z pamięcią operacyjną 128 k i 4 jednostkami pamięci dyskowej.

W międzyczasie filialny ośrodek w Ełku otrzymał 2 minikomputery MERA serii 300, które do czasu zainstalowania tam komputera R-32 będą stanowiły przedmiot zadań programowych tego ośrodka.

Dla zilustrowania dynamicznego rozwoju ZETO w Białymstoku warto przytoczyć nieco danych statystycznych. O sprzęcie już było wyżej. Kadra ośrodka, abstrahując od strony jakościowej, notowała spory przyrost ilościowy. Największy datuje się w okresie 1972—74 roku, a więc w oczekiwaniu na instalację komputera. Stan zwiększył się wówczas ze 101 do 190 osób. Obecnie ZETO zatrudnia 236 osób (w tym jednak sporą grupę perforatorek albowiem większość użytkowników ZETO musi obsługiwać kompleksowo — od przygotowania danych począwszy) a jest to z reguły kadra młoda lecz dobrze przeszkolona. Sporą pomocą w pracy szkoleniowej przychodzi NOT i PTE.

Wartość sprzedaży usług informatycznych przyrastała równie dynamicznie. Od 25 tys. zł w roku 1967 do 16 626 tys. zł w roku 1974, ca 40 mln zł w roku 1976.

Z ważniejszych, eksploatowanych obecnie systemów należy wymienić:

- system inkasa należności za gaz i energię elektryczną eksploatowany dla Zakładów Energetycznych
- system gospodarki materiałowej dla ZPB Fasty, Zakładów Energetycznych, Zakładów Wytwórczych Osprzętu Kablowego w Ełku, Przedsiębiorstwa Budownictwa Przemysłowego, Okręgowego Przedsiębiorstwa Przemysłu Zbożowo-Młynarskiego, WSS Społem, Przedsiębiorstwa Robót Inżynieryjno-Montażowych Budownictwa Rolniczego
- system gospodarki wyrobami gotowymi — dla Wojewódzkich Przedsiębiorstw Przemysłu Mięsnego, Wytwórni Wódek, BZPB Fasty
- system TEMIDA dla KWMO
- system obliczania pikiet tachimetrycznych — dla Biura Projektów Wodnych i Melioracyjnych
- system finansowo-księgowy — dla Zakładów Produkcji Maszyn Handlowych, Fabryki Narzędzi i Uchwytów, NBP i PKO oraz Warszawskich Zakładów Telewizyjnych
- system gospodarki transportowej — dla Zakładów Energetycznych.

Nie jest to oczywiście kompletna lista użytkowników i eksploatowanych systemów. Jej właściwy wymiar świadczyłby efektowniej z jednej strony o możliwościach białostockich informatyków w sferze produkcji i eksploatacji oprogramowania oraz obsługi sprzętu, z drugiej o skutecznej propagandzie korzyści płynących z ETO wśród białostockich przedsiębiorstw.

Suma tych osiągnięć wystawia dobre świadectwo białostockiemu ZETO w dniu jego jubileuszu.

Krystyn Bernatowicz

FOTO 2. W ramach uroczystości 10-lecia ZETO odbyło się seminarium naukowe i akademia, w czasie której wręczono zasłużonym pracownikom odznaczenia państwowe (1 złoty, 4 srebrne i 3 brązowe Krzyże Zasługi) i regionalne. Na zdjęciu dyrektor ZETO w Białymstoku mgr inż. Witold Szyszło składa gratulacje odznaczonemu Srebrnym Krzyżem Zasługi mgr inż. Aleksandrowi Wydryckiemu (wicedyrektorowi ZETO). Szkoda, że obchody jubileuszu ZETO w Białymstoku zbiegły się z terminem posiedzenia kierownictwa Zjednoczenia Informatyki. Należy sądzić, że w innym terminie prezentacja Zjednoczenia Informatyki na uroczystości jubileuszowe byłaby jak zazwyczaj na jubileuszach innych Ośrodków ZETO, znacznie szersza.

MERA-303 w szczecińskiej szkole

Jak informuje Ekspress Wieczorny z dnia 6 grudnia 1976 r. w notatce sygnowanej literami (WIT) w Zespole Szkół Ekonomicznych nr 1 w Szczecinie oddano do użytku pracownię techniki obliczeniowej wyposażoną w minikomputer MERA-303 oraz szereg maszyn małej i średniej mechanizacji. Z pracowni tej będą korzystać uczniowie czwartych klas maturalnych i studium pomaturalnego. Jest to pierwsza tego typu pracownia szkolna w woj. szczecińskim.

W roku przyszłym minikomputer MERA-303 zostanie wyposażony w monitor ekranowy i pamięć dyskową. Pracownia powstała z inicjatywy nauczyciela Romualda Czurłowskiego a została wyposażona z pomocą Kuratorium Oświaty i Wychowania, Komitetu Rodzicielskiego, wydziału zaocznego i samorządu uczniowskiego.

Informację tę zamieszczamy bez komentarzy i oczekujemy od czytelników sygnałów o podobnych inicjatywach.

(P)

Międzynarodowa Konferencja: Zastosowanie informatyki w przedsiębiorstwach przemysłu maszynowego COMPCONTROL-77

W dniach 14—18 listopada 1977 r., w salach NOT przy ul. Czackiego 3/5 w Warszawie odbędzie się Międzynarodowa Konferencja na temat: Zastosowanie informatyki w przedsiębiorstwach przemysłu maszynowego COMPCONTROL-77. Celem konferencji COMPCONTROL-77 jest dokonanie aktualnego przeglądu osiągnięć w zakresie zweryfikowanych w praktyce, zastosowań komputerów do przetwarzania danych w zarządzaniu i automatyzacji prac inżynierskich, jak również wymiana informacji i doświadczeń oraz prezentacja nowych koncepcji opracowania systemów informatycznych zarządzania oraz sterowania procesami wytwarzania.

Międzynarodowe konferencje zastosowań informatyki w przedsiębiorstwach przemysłu maszynowego pod nazwą COMPCONTROL organizowane były od roku 1968 przez węgierski Gepipari Tudomanyos Egyesület.

Podczas IV narady Przewodniczących i Sekretarzy Stowarzyszeń Naukowo-Technicznych Budowy Maszyn Krajów RWPG, która odbyła się w październiku 1975 r. w Scheibenmuhle (NRD) ustalono, że w 1977 r. międzynarodowa konferencja COMPCONTROL-77, będzie zorganizowana przez Stowarzyszenie Inżynierów i Techników Mechaników Polskich SIMP w Warszawie, a w latach następnych przez inne kraje członkowskie RWPG.

Realizując te ustalenia, w marcu 1976 r., Zespół Informatyki przy Zarządzie Głównym SIMP w Warszawie, powołał Komitet Organizacyjno-Programowy Konferencji COMPCONTROL-77.

Obrady odbywać się będą w dwóch sekcjach problemowych:

zastosowania komputerów w zarządzaniu oraz zastosowania komputerów w pracach inżynierskich.

W sekcji I, **Zastosowania komputerów w zarządzaniu,** przedmiotem dyskusji będą następujące referaty:

- strategia rozwoju i wdrażania systemów informatycznych w przedsiębiorstwach
- doświadczenia projektowania i wdrażania systemów informatycznych w przedsiębiorstwach

— wsparte komputerem zautomatyzowane systemy zarządzania i sterowania
— zintegrowane systemy zarządzania
— programowanie produkcji
— operatywne i warsztatowe sterowanie produkcją

- przygotowanie kadry użytkowników, szczególnie kadry kierowniczej i problemy socjalno-psychologiczne wdrażania systemów
- dialog człowieka z komputerem
- zastosowanie typowych pakietów

— zagadnienia banku danych
— zastosowanie typowych pakietów programów zorientowanych na problemy zarządzania przedsiębiorstwem

- organizacja i przygotowanie danych
- przygotowanie techniczno-organizacyjne obiektów.

W sekcji II: **Zastosowanie komputerów w pracach inżynierskich** będą omówione referaty:

- wspomagane komputerem projektowania konstrukcji

— analiza wytrzymałości i dynamiki konstrukcji metodami modelowania cyfrowego
— optymalizacja konstrukcji
— grafika komputerowa i przetwarzanie kształtów
— organizacja systemów projektujących wspomaganych komputerem

- wspomagane komputerem projektowanie technologii

— projektowanie procesów technologicznych dla obrabiarek konwencjonalnych
— komputerowe systemy programowania i sterowania obrabiarek i urządzeń technologicznych

- automatyzacja badań wyrobów
- zintegrowane systemy informatyczne technicznego przygotowania produkcji (integracja projektowania konstrukcji i technologii).

W czasie konferencji zostanie wygłoszonych około 20 referatów głównych przez przedstawicieli z Węgier, Związku Radzieckiego, Bułgarii, Polski i innych oraz około 40 dodatkowych referatów.

Wszystkie materiały konferencyjne (referaty główne i dodatkowe) będą wydane w dwóch zestawach:

— zestaw pierwszy, obejmuje wszystkie referaty główne i dodatkowe opracowane w jednym z trzech obowiązujących podczas konferencji języków: rosyjskim, niemieckim lub angielskim. W zestawie tym będą umieszczone skróty referatów we wszystkich trzech językach
— zestaw drugi, obejmie ten sam zbiór referatów w języku polskim.

Zestaw drugi przeznaczony jest dla uczestników konferencji zgłoszonych przez przedsiębiorstwa naszego kraju.

Podczas obrad, zapewnione będą tłumaczenia symultaniczne. W marcu 1977 r. zostaną wysłane informatory o wymienionej konferencji wraz z kartami zgłoszeń, do zainteresowanych jednostek organizacyjnych przemysłu maszynowego, przemysłu maszyn ciężkich i rolniczych, hutnictwa itp.

Dla lepszego zorientowania organizatorów konferencji w zakresie zainteresowań organizowaną konferencją, pożądane jest wcześniejsze awizowanie zainteresowań poszczególnych jednostek organizacyjnych.

Korespondencję w sprawach związanych z organizowaną konferencją należy kierować na adres: SEKRETARIAT KONFERENCJI COMPCONTROL-77, Ośrodek Doskonalenia Kadr Stowarzyszenia Inżynierów i Techników Mechaników Polskich, ul. Mickiewicza 9, 01-517 Warszawa.

Czytajcie i prenumerujcie Informatykę

Tydzień Techniki Bułgarskiej w Polsce

W dniach od 2 do 6 listopada 1976 r. odbywały się sympozja zorganizowane z okazji Tygodnia Techniki Bułgarskiej w Polsce zorganizowanego przez Centralną Radę Stowarzyszeń Naukowo-Technicznych w Bułgarii i Naczelną Organizację Techniczną w Polsce.

W Warszawie, Gdańsku i Skierniewicach wygłoszone zostały referaty techniczne, wzbogacane projekcjami filmowymi. W Muzeum Techniki w Warszawie czynna była wystawa fotogramów przedstawiających osiągnięcia przemysłu i gospodarki bułgarskiej.

Wśród wygłoszonych w czasie sympozjów referatów znalazły się również cztery, które dotyczyły informatyki a mianowicie:
— pamięci zewnętrzne dla elektronicznych maszyn cyfrowych
— projektowanie i wdrażanie systemów automatyzacji procesów technologicznych w przemyśle chemicznym
— technologia MOS i układy scalone MOS
— obrabiarki i zespoły maszyn sterowanych numerycznie.

Przygotowania do produkcji sprzętu komputerowego podjęto w Bułgarii dopiero w roku 1965. W ciągu zaledwie pięciu lat przygotowano i podjęto produkcję sprzętu, przede wszystkim pamięci zewnętrznych dla potrzeb Jednolitego Systemu EMC. Chociaż niektóre z nich są już dobrze znane na naszym rynku, interesujące było podanie kompletnego przeglądu charakterystyk wszystkich modeli pamięci zewnętrznych aktualnie produkowanych przez przemysł bułgarski. A oto ich skrótowy przegląd.

PAMIĘCI TAŚMOWE

Model 5012

Podstawowym modelem jest EC 5012-01 dostosowany do współpracy z komputerami JS EMC o następujących parametrach technicznych:
— szybkość przesyłania informacji — 64 k bajtów/s
— metoda zapisu — NRZ1
— gęstość zapisu — 8 lub 32 rządki/mm
— liczba ścieżek — 9
— gabaryty — 900 × 700 × 1800 mm
— ciężar — poniżej 450 kg
— średni czas pomiędzy uszkodzeniami — większy niż 400 godzin.

Oprócz powyższego modelu produkowane są jego dodatkowe wersje, a mianowicie:
— IZOT 5012 11-01 przeznaczona dla komputerów MIŃSK-32
— IZOT 5012 0-01 przeznaczona dla komputerów serii ODRA 1300.

Modele 5003, 5005, 5006

Dla potrzeb małych komputerów produkowane są trzy modele wolnych pamięci taśmowych, a mianowicie:

● **IZOT 5003 E**
— szybkość przesyłania informacji — 10 k bajtów/s
— metoda zapisu — NRZ 1
— gęstość zapisu — 32 rządki/mm
— liczba ścieżek — 9
— gabaryty — 520 × 318 × 493 mm
— średni czas pomiędzy uszkodzeniami — 1000 godzin

● **IZOT 5006 E oraz 5005 E**
— szybkość przesyłania informacji — 32 k bajtów/s
— metoda zapisu — NRZ 1
— liczba ścieżek — 9
— gabaryty — 610 × 483 × 460 mm

Model 5005 E jest wariantem modelu 5006 E różniącym się wyłącznie mniejszą szybkością przesyłania informacji (20 k bajtów/s).

PAMIĘCI DYSKOWE

Model EC 5052

Chronologicznie pierwszy model, przeznaczony do współpracy z komputerami JS EMC (poprzez jednostkę sterującą EC 5552).

Charakterystyka techniczna:
— nośnik — pakiet wymienny 6-dyskowy EC 5053 (10 powierzchni roboczych) o pojemności 7,25 M bajtów
— szybkość przesyłania informacji — 156 k bajtów/s
— gęstość zapisu — 30—45 bitów/mm
— średni czas ustawiania głowicy — 60 ms
— szybkość obrotowa dysku — 2400 obr./min
— gabaryty — 610 × 775 × 975 mm.

Model EC 5061

Model ten przeznaczony jest również do współpracy z komputerami JS EMC (poprzez jednostkę sterującą EC 5561).

Charakterystyka techniczna:
— nośnik — pakiet wymienny 11-dyskowy EC 5161 (20 powierzchni roboczych) o pojemności 29 M bajtów
— szybkość przesyłania informacji — 312 k bajtów/s
— gęstość zapisu — 60—70 bitów/mm
— szybkość obrotowa dysku — 2400 obr./min
— średni czas ustawiania głowicy — 50 ms
— gabaryty — 610 × 775 × 975 mm
— ciężar — 180 kg.

Model IZOT 1370

Dostosowany do współpracy z małymi i średnimi komputerami.

Charakterystyka techniczna:
— nośnik — dysk stały o pojemności 3 M bajtów oraz wymienny dysk kasetowy typ EC 5269 A o pojemności 3 M bajtów
— szybkość przesyłania informacji — 312 k bajtów/s
— gęstość zapisu — 60—90 bitów/mm
— szybkość obrotowa dysku — 2400 obr./min
— średni czas ustawiania głowicy — 45 ms
— gabaryty — 264 × 480 × 765 mm.

Model IZOT 5971 E

Przeznaczony głównie do zbierania i wyprowadzania danych.

Charakterystyka techniczna:
— nośność — dysk elastyczny typu IBM Diskette o pojemności 320 k bajtów (1 powierzchnia robocza)
— szybkość przesyłania informacji — 250 k bitów/s
— gęstość zapisu — 126 bitów/mm
— szybkość obrotowa dysku — 360 obr./min
— czas dostępu do ścieżki — poniżej 20 ms
— gabaryty — 135 × 325 × 380 mm.

Wzrastające wymagania w zakresie pojemności pamięci i zmniejszania czasu dostępu do informacji, zmuszają do opracowywania nowych i bardziej doskonałych modeli pamięci zewnętrznych. W Bułgarskiej Republice Ludowej opracowuje się obecnie ulepszony model jednostki pamięci taśmowej ES 5013 o prędkości przesuwu taśmy 5 m/s i gęstości zapisu 63 rządki/mm. Pozwoli to podwoić ilość zapamiętanych informacji i zmniejszyć 2,5-krotnie czas dostępu w porównaniu z obecnie produkowanym modelem 5012. W dziedzinie pamięci dyskowych opracowuje się nowy model o pojemności 200 M bajtów oraz jednostkę podwójną o pojemności 400 M bajtów a także jednostki sterujące umożliwiające współpracę tych modeli z komputerami JS EMC.

Andrzej Klimek

Kartka z Finlandii

Finlandia reklamuje się turystom jako kraj reniferów i polarnego, niezachodzącego słońca. Można jednak zauważyć, że obok dziewiczej i pięknej natury rozwija się tu nowoczesna gospodarka, co oczywiście odbywa się nie bez udziału komputeryzacji.

W Finlandii działa w tej chwili ponad 50 różnych firm komputerowych producentów i dostawców sprzętu i oprogramowania, firm serwisowych, konsultacyjnych i marketingowych. Ogólny obraz sytuacji jest jednak zaciemniony przez fakt, że większość tych firm reprezentuje jedynie handlowe interesy potentatów zagranicznych i to głównie amerykańskich. Niekiedy jedyną działalnością tego rodzaju firm jest dolepianie na urządzeniach dostarczanych lokalnemu klientowi, a sprowadzanych z zagranicy, etykietki

z własną nazwą. Niektóre filie znanych koncernów prowadzą produkcję oprogramowania na użytek lokalnego rynku.

Najciekawsze dla nas są oczywiście konstrukcje oryginalne i ich wytwórcy. W Finlandii wytwarza się minikomputery, terminale, modemy i szereg innych urządzeń teletransmisyjnych.

Najbardziej aktywną firmą jest STRÖMBERG — fabryka o bogatych tradycjach w wytwarzaniu urządzeń energetycznych i elektronicznej aparatury przemysłowej. Na początku lat siedemdziesiątych opracowano w niej, z myślą o wytwarzanych na miejscu systemach automatycznego sterowania, minikomputer STRÖMBERG 1000.

Obecnie oferuje inny, modularny system przetwarzania sygnałów i sterowania STRÖMBERG 120, wykorzystujący jednostkę centralną maszyny „NOVA" (DATA GENERAL CORP.). Drugim znanym producentem jest NOKIA ELEKTRONIIKKA. Firma ta produkuje sprzęt teleprzetwarzania, a zwłaszcza procesory telekomunikacyjne, urządzenia teletransmisji i terminale. Dostarcza ona kompletne systemy wieloodstępne do obsługi domów towarowych, magazynów itp.

Liczba zainstalowanych komputerów w Finlandii rośnie bardzo szybko. Jak podaje rocznik fińskiego Stowarzyszenia Informatyków [1]), według stanu z 1 stycznia 1976 roku, działało w Finlandii ponad 200 ośrodków obliczeniowych, a w eksploatacji znajdowało się 1079 komputerów. Jest to o 425 egz. więcej niż w roku poprzednim. Z tego połowę stanowią maszyny typu DATA GENERAL NOVA.

Spośród większych komputerów najpopularniejszym modelem jest IBM S/370/135 (31 egzemplarzy). Koncern IBM jest jednocześnie dostawcą największej liczby różnych maszyn (224 egzemplarze). Drugim w kolejności dostawcą jest HONEYWELL-BULL (98 egzemplarzy). Park komputerowy jest w Finlandii bardzo zróżnicowany i pochodzi od 25 producentów. W Helsinkach funkcjonuje między innymi maszyna Jednolitego Systemu EC 1022, która jest własnością radziecko-fińskiej spółki ELORG-DATA. Firma ta, „lansuje" na rynku fińskim komputery Jednolitego Systemu.

Wraz z trudnościami producentów sprzętu pojawiają się trudne problemy u ich klientów. Na przykład w poważnym kłopocie znaleźli się ostatnio użytkownicy maszyn szwedzkiej firmy DATASAAB. Po fuzji tej firmy z amerykańską firmą UNIVAC, nowa spółka — DATASAAB-UNIVAC — nie produkuje maszyn rodziny DATASAAB D-20. Użytkownicy tych maszyn z całym zestawem oprogramowania zostali „na lodzie". Sytuacja jest o tyle nietypowa, że głównym językiem zewnętrznym maszyn tej rodziny był ALGOL-GENIUS, język będący szczególnym rozszerzeniem ALGOLU 60, z możliwościami przetwarzania danych masowych o różnorodnych strukturach. Powstała spółka zapewnia wprawdzie, że byłym użytkownikom maszyn DATASAAB dostarczać będzie komputery UNIVAC z translatorem języka ALGOL-GENIUS, jednak nie wiadomo, czy można na to liczyć i czy rozwiązanie to okaże się użyteczne.

W Finlandii istnieje (w Polsce nie istnieje — przyp. red.) Stowarzyszenie Informatyków. Liczy ono obecnie prawie 6 tys. członków, a jego działalność obejmuje: wydawanie fachowych periodyków (10 pozycji), organizowanie i inicjowanie nowych form szkolenia zawodowego, pewne prace badawcze oraz prace z dziedziny standaryzacji. Stowarzyszenie Informatyków organizuje także międzynarodowe konferencje informatyków i aktywnie wpływa na rozwój informatyki w tym kraju.

Oprac. Antoni Wolski

[1]) ATK — vuosikirja. Tietojenkäsittelyliitto Ry, Helsinki 1976. Z podanej publikacji zaczerpnięto wszystkie dane liczbowe.

Informatyka po angielsku

Nowoczesność w gospodarce zapasami

Przedsiębiorstwo, o którym będzie mowa, chłonie nową technikę w większym stopniu, aniżeli przeciętne przedsiębiorstwo angielskie. Wystarczy powiedzieć, że jako pierwsze w Wielkiej Brytanii zastosowało komputery serii 6000 HONEYWELLA, jako drugie postawiło centralny, zautomatyzowany magazyn oraz należy do grupy najbardziej zaawansowanych użytkowników mikrofilmowych urządzeń wyjściowych (C.O.M.) i jako jedno z nielicznych może się poszczycić niemal kompleksowym zespołem systemów informatycznych, bo obejmującym 80—90% wszystkich problemów zarządzania. Jego doświadczenia zasługują na uwagę.

W warunkach kapitalistycznej konkurencji dostawca gromadzi towary, aby móc pokryć każde zapotrzebowanie i racjonalna gospodarka tymi zapasami jest dla niego problemem pierwszorzędnej wagi. Stosuje w tym celu środki techniczne i organizacyjne, które mogą znaleźć interesujące zastosowanie także w przedsiębiorstwach socjalistycznych, tym bardziej, że wbrew rozpowszechnionym u nas opiniom zaopatrzenie materiałowe nie jest bynajmniej łatwe ani bezproblemowe.

Wprawdzie szereg materiałów trafia bezpośrednio na miejsca ich użycia w terminie wynikającym z harmonogramu robót, z pominięciem magazynów, ale do rangi problemu urasta wybór dostawców, gwarantujących należytą terminowość, jakość i niską cenę.

EASTERN ELECTRICITY, bo o nim tu mowa, zaopatruje się w materiały do eksploatacji sieci elektrycznych, do robót instalatorskich u odbiorców oraz w odbiorniki elektryczne dla potrzeb gospodarstw domowych, sprzedawane następnie w 120 sklepach.

Przedsiębiorstwo to prowadzi 2 magazyny centralne i 31 pomocniczych oraz eksploatuje około 600 samochodów, z których każdy traktowany jest w systemie informatycznym jako oddzielny magazyn. Dzieje się tak dlatego, że są to głównie samochody pogotowia technicznego i brygad instalatorskich (w których znajduje się „żelazny zapas" materiałów niezbędnych w codziennej działalności), a ponadto samochody przystosowane do transportu kontenerów, używane do rozwożenia materiałów i towarów.

W centralnym, skomputeryzowanym biurze księgowości funkcjonuje sekcja, licząca około 30 osób, której zadaniem jest organizacja procesu przetwarzania informacji i obiegu dokumentów.

Sekcja przetwarza dokumenty transakcyjne, których liczba waha się od 80 000 do 150 000 tygodniowo. Dotyczą one 16 500 różnych pozycji magazynowych. Sekcja posługuje się dwoma systemami informatycznymi, z których jeden obejmuje gospodarkę zapasami, drugi — rozliczenia z dostawcami. Oba systemy opierają się na wspólnej bazie danych, obejmującej także pozostałych dziesięć systemów. Dużym uproszczeniem, a zarazem istotnym krokiem ku integracji systemów jest opracowanie i zastosowanie do potrzeb wszystkich wspomnianych systemów tak zwanego zintegrowanego systemu wprowadzania danych, pozwalającego na wprowadzanie do maszyny danych wejściowych w dowolnej kolejności i dotyczących każdego z dwunastu podstawowych systemów informatycznych zarządzania, w tym także obu systemów związanych z gospodarką zapasami.

Zintegrowany system wprowadzania danych uprościł przede wszystkim pracę stacji przygotowania danych, która korzystając z rejestratorów taśmowych i dyskowych, może zapisywać na wspólnej szpuli taśmy magnetycznej dane źródłowe należące do różnych systemów.

Można nawet powiedzieć, że opracowanie takiego systemu stało się koniecznością wynikająca z zastosowania nowoczesnych urządzeń do rejestracji danych na maszynowym nośniku informacji.

Innym, godnym uwagi, nowoczesnym środkiem technicznym, który znalazł szerokie zastosowanie w systemie gospodarki zapasami, było mikrofilmowe urządzenie wyjściowe, znane pod angielską nazwą COMPUTER OUTPUT MICROFILM. Urządzenie to pozwala rejestrować dane na błonie filmowej, ciętej następnie na fiszki, z których każda zawiera 207 pól roboczych, jedno funkcyjne i dwa wiersze nagłówka, czytelne gołym okiem. Dane zredagowane przez komputer wprowadza się na taśmę magnetyczną, skąd następnie, już poza maszyną, przenoszone są na mikrofilm. Pozwala to pominąć drukarkę wierszową. Mikrofilm czytelny jest przy użyciu specjalnych czytników, których ponad dwieście zainstalowano w poszczególnych biurach i sklepach oraz w magazynach, czyniąc z nich urządzenie codziennego użytku.

W systemie gospodarki zapasami mikrofilm znalazł zastosowanie między innymi do prowadzenia katalogów materiałowych. Cały katalog mieści się na czterech fiszkach, jest co tydzień aktualizowany i rozsyłany, po skopiowaniu, do wszystkich zainteresowanych komórek przedsiębiorstwa. Dzięki temu katalog ten może być na bieżąco aktualizowany, a posługiwanie się nim jest proste i szybkie. Oszczędza się nie tylko papier, ale także czas, jakiego wymagało manipulowanie wielostronicowymi wydrukami. Bądź, co bądź, na jednej fiszce mieści się treść 207 arkuszy papieru tabulogramowego!

Mikrofiszka zawiera dane ilościowo-wartościowe poszczególnych pozycji katalogowych, co znacznie zwiększa użyteczność katalogu.

Numer katalogowy materiału jest łatwy do odczytania: dwie pierwsze cyfry oznaczają grupę materiałów, 3 następne kolejny numer materiału w grupie. Na ostatniej pozycji, szóstej, znajduje się litera kontrolna. Oryginalnym, a jednocześnie niezwykle prostym rozwiązaniem problemu kodu materiałowego jest wykorzystanie alfanumerycznej nazwy materiału, składającej się z kilku wyrazów o łącznej liczbie znaków nie przekraczającej 24.

Nazewnictwo materiałów jest w taki sposób uporządkowane, że w miarę potrzeby, dokonuje się sortowań opierając się na nazwach materiałów.

System gospodarki zapasami zawiera około 40 różnych zestawień [1]), z których część dotyczy organizacji procesu przetwarzania danych, ale znaczna większość, ukazując się w różnych terminach i docierając do różnych komórek organizacyjnych i szczebli zarządzania sprawia, że kierownictwo przedsiębiorstwa może włączyć do swego programu pracy codzienne wykrywanie rezerw i codzienne, systematyczne ich zagospodarowywanie. W ten sposób realizuje się jedno z ważniejszych źródeł efektywności systemów informatycznych.

Ciekawym i ważnym elementem systemu gospodarki zapasami jest para magazynów centralnych, zlokalizowanych na przeciwległych krańcach okręgu energetycznego. W połowie 1973 roku oddano do eksploatacji drugi z tych magazynów, w pełni zautomatyzowany. Tym samym zakończono wprowadzanie nowoczesnego, kompleksowo rozwiązanego systemu magazynowania i dystrybucji materiałów i towarów.

W skład tego systemu wchodzą oprócz wspomnianych magazynów, także:

— kontenery

— samochody do przewozu kontenerów

— dwie bazy techniczne transportu, zlokalizowane przy magazynach centralnych

— system łączności dalekopisowej

— system informatyczny gospodarki magazynowej (omówiony poprzednio)

— nowo utworzony wydział (pion) zaopatrzenia, powołany do koordynacji zaopatrzenia i gospodarki zapasami.

Lokalizacja magazynów centralnych nie jest przypadkowa. Organizatorzy systemu ubolewają, że władze miejskie Londynu nie znalazły odpowiedniego miejsca na magazyn, a niewielkie odchylenie od wyznaczonego optymalnie powoduje wzrost kosztów dystrybucji o 500 £ rocznie na milę!

Magazyn zlokalizowany na północnym krańcu Londynu ma pojemność około 2 milionów stóp kubicznych, kosztował 1250 tysięcy funtów i przeznaczony jest do współpracy z 1/3 obszaru przedsiębiorstwa, a ze względu na dużą gęstość zaludnienia tych terenów, obejmuje 2/3 wszystkich materiałów i towarów przedsiębiorstwa. Główną część magazynu stanowią wysokie na 65 stóp pomieszczenia dostosowane do przechowywania palet z towarami. Pracuje tu 5 automatycznych dźwigów, sterowanych przez komputer, który optymalizuje wykorzystanie powierzchni magazynowej i w taki sposób kieruje pracą dźwigów, aby uzyskać jak największą przelotowość, a jednocześnie zmniejszyć zużycie energii i zaoszczędzić pracę dźwigów.

Między innymi w tym celu, stosownie do sezonu, komputer przeprowadza operację przenoszenia materiałów bardziej chodliwych na niższe półki, a mniej chodliwych, na półki wyższe.

Wydawanie materiałów i towarów odbywa się na podstawie zamówień nadesłanych dalekopisem z biur poszczególnych rejonów. Zamówione towary dostarczane są do odległych nawet rejonów w ciągu jednego dnia.

Zamawiający ma obowiązek wskazania kolejności rozładunku materiałów, z których część dostarczana jest wprost do mieszkań klientów. Stosownie do tych wskazówek, komputer sporządza zbiorcze zestawienia zamówień, a następnie ustala kolejność ładowania kontenerów w taki sposób, aby w trakcie ich rozładowywania nie było potrzeby przeładowywania.
Pracując na podstawie zestawionej listy zamówień komputer wybiera palety, stosując między innymi następujące kryteria:

— wszystkie dźwigi (w liczbie 5) powinny być wykorzystane jednocześnie

— palety częściowo zapełnione powinny być opróżnione w pierwszej kolejności

— palety umieszczone w komorach ładunkowych magazynu, większych niż to niezbędne, powinny być opróżnione w pierwszej kolejności, to znaczy przed paletami znajdującymi się w komorach wykorzystanych w całości

— paleta z urządzeniem, ma pierwszeństwo przed paletą zawierającą kilka różnych urządzeń (materiałów).

Zakończenie prac ładunkowych i odesłanie palety na miejsce, jeśli nie jest pusta, jest dla komputera sygnałem do zaktualizowania zbiorów. Po aktualizacji każda transakcja zostaje zarejestrowana na taśmie dziurkowanej.

W głównej części magazynu, gdzie operuje 5 dźwigów i gdzie człowiek praktycznie nie ma dostępu, można przechowywać około 20 000 urządzeń wielkości dużej lodówki. Magazyn przechowuje 250 typów większych odbiorników elektrycznych, mających zastosowanie w gospodarstwie domowym, 300 typów drobnych urządzeń, 950 typów materiałów do prac technicznych, 500 typów materiałów instalatorskich i około 6000 typów części zamiennych.

Obok magazynów pobudowano centralne bazy transportu, mając na uwadze to, że głównymi dysponentami środków transportowych są magazyny centralne. Wprowadzenie kompleksowego systemu magazynowania i dystrybucji pozwoliło zmniejszyć wartość zapasów z 6,5 do 3,5 miliona funtów szterlingów.

W ogólnej ocenie przedsięwzięcia zwraca uwagę fakt, że zastosowanie informatyki w powiązaniu z organizacją wykorzystującą możliwości stwarzane przez informatykę, przekształciła dawne systemy ewidencyjne w system, pozwalający oceniać na bieżąco sytuację gospodarczą przedsiębiorstwa i pozwalający kierować jego pracą, tak aby zapewnić stały postęp ekonomiczny.

JANUSZ ILCZUK
Warszawa

---

[1]) Szersze omówienie tematu znaleźć można w referacie opublikowanym w materiałach na kursokonferencję naukową na temat: Organizacja gospodarki materiałowej w przedsiębiorstwie przemysłowym w warunkach zastosowania elektronicznej techniki obliczeniowej, zorganizowaną w Warszawie, w 1976 r. przez Towarzystwo Naukowe Organizacji i Kierowania — Oddział Warszawski, Stowarzyszenie Księgowych w Polsce — Oddział Warszawski i Stowarzyszenie Inżynierów i Techników Mechaników Polskich.

Czy język ojczysty będzie językiem informatyków?

„Język jako narzędzie porozumiewania się jest neutralny, jak każde narzędzie.
Od jednostek posługujących się nim będzie zależało, jak będzie ono wykorzystane".
Prof. M. Szymczak [1]

Zarówno możliwość, jak i szczególnie efektywność korzystania z określonej dziedziny wiedzy w dużym stopniu zależy od komunikatywności przedstawianych składników danej rzeczywistości, czyli od zrozumiałości języka, jakim posługują się jej przedstawiciele. Niestety, dezintegracja języka pogłębia się; już nie tylko poszczególne kręgi środowiskowo-zawodowe, ale każda dyscyplina naukowa posługuje się coraz bardziej hermetycznym zasobem słów, terminów i pojęć. Stąd też nie tylko laik, ale nawet specjalista jednej dziedziny naukowej nie rozumie dziś tego, co mówi specjalista innej dziedziny; chemik nie rozumie fizyka, fizyk — elektronika, a ten znów ekonomisty i tak dalej, i tak dalej.

To ujemne zjawisko narasta i pogłębia się, a skutki są coraz bardziej odczuwalne i coraz bardziej niepokoją opinię społeczną [2], a tym szczególniej, że — jak pisze w „Expresie Wieczornym" popularny felietonista E. WIECZ [3] — „Jednocześnie z lawinowo narastającym owym hermetycznym językiem naukowym, pojawia się — na szczęście jako zjawisko marginesowe — skłonność do ubierania w pseudonaukową terminologię spraw z dziedzin prostych, potocznych, powszechnie znanych.

Zabiegi te służą do nobilitowania rzeczy oczywistych przez przydawanie im pozorów naukowych".

Nie lepiej, a być może nawet gorzej jest z tzw. językiem informatyków. Obok bowiem prób pseudonaukowości w kwestiach terminologii informatycznej [4] oraz błądzenia po meandrach językoznawstwa [5], występuje zjawisko uchylania się zobowiązanych od podejmowania kwestii terminologicznych i przechodzenia nad tym problemem do porządku [6] lub lawinowego stosowania wyrazów z gruntu niezrozumiałych nawet dla szerszego kręgu informatyków [7].

Podkreślić należy, że zagadnienia terminologiczne trudne są właśnie w informatyce.

Wynika to z trzech zasadniczych przyczyn, w tym z dwóch o charakterze ogólnym i z trzeciej, mającej charakter szczególny.

Po pierwsze. Informatyka, jako dziedzina wiedzy i działalności ludzkiej, obejmująca przekształcanie informacji za pomocą techniki obliczeniowej, w sferze nauki ma charakter interdyscyplinarny, a w sferze praktyki ma charakter integracyjny celowej działalności różnych służb (np. kierowniczej, organizatorskiej, planistycznej, rachunkowości, INTE, ETO, normalizacyjnej i in.), tak w odniesieniu do poszczególnych jednostek organizacyjnych (informacja wewnętrzna), jak i większych organizmów gospodarczych i całej gospodarki narodowej (informacja zewnętrzna).

Zatem — **język informatyków** nie powinien lecz musi być szeroko **zrozumiały**; nie może więc być językiem wąskiej specjalizacji, o trudnej lub wręcz niezrozumiałej terminologii dla użytkowników-nieinformatyków.

Po drugie. Informatyka pozostaje nadal dziedziną wiedzy względnie młodą i ciągle nie posiada ustalonego słownictwa (terminologii) [8].

Zdarza się więc, że jedno pojęcie bywa określane za pomocą kilku różnych słów (nazw) lub, co gorsza różnych rzeczy lub pojęć: a już najgorzej, gdy odmienne terminy dla odmiennych pojęć identyfikowane są przemiennie ze sobą.

I po trzecie. Na gruncie polskim, w warunkach gdy w zakresie systemów przetwarzania informacji, tak w sferze materialnej, jak i w sferze logicznej (oprogramowanie) — językiem wyjściowym nie jest język ojczysty, lecz angielski, trudności terminologiczne potęgują się, a niekomunikatywność języka informatycznego rośnie nieomal w postępie geometrycznym. Źródłem tego zjawiska jest nie tylko nie w pełni jeszcze ustabilizowana terminologia w podstawowym języku, za jaki przyjmuje się w informatyce język angielski, ale też:

— utrzymanie dla niektórych rzeczy lub pojęć terminów w brzmieniu angielskim, bez próby tłumaczenia znaczenia tych wyrazów w języku polskim

— zastępowanie niektórych określeń angielskich wyrazami polskimi, ale z pomijaniem utartych znaczeń pojęciowych tych ostatnich

— „spolszczanie" na siłę słów obcojęzycznych i tworzenie dziwolągów językowych.

Jak można wnioskować z różnych publikacji i jak wynika z artykułów zamieszczonych na łamach INFORMATYKI, problemy językowe informatyki nurtują środowisko samych informatyków.

Jednakże z głębokim odczuciem niewłaściwości istniejącego stanu i potrzeby jego zmiany nie zawsze idzie w parze rzetelna wiedza, a tym samym realność wnioskowania w tym zakresie [9].

Wobec takiego stanu i mając na uwadze, że problemy słownictwa informatycznego będą przedmiotem dyskusji i kompleksowych opracowań, wydaje się konieczne przytoczenie, na użytek wszystkich zainteresowanych informatyków, wypowiedzi dwóch osób kompetentnych — każda w swoim zakresie.

Profesor M. Szymczak, dyrektor Instytutu Języka Polskiego UW i przewodniczący Komisji Kultury Języka PAN, w rozmowie z przedstawicielką redakcji „Polityki" m. in. tak powiedział: „Każdy język w inny nieco sposób dzieli i klasyfikuje otaczającą nas rzeczywistość, w inny sposób wyodrębnia jej fragmenty, nadając im nazwy. Nic więc dziwnego, że zakres znaczeniowy wyrazów jednego języka nie pokrywa się ściśle z zakresem znaczeniowym innego języka (...) język angielski (...) wyodrębnił jako przedmiot myśli społecznej takie fragmenty rzeczywistości, które nie zostały nazwowo wyodrębnione, np. w języku polskim (...).

Język polski ma również wiele pojęć i wyrazów określających te pojęcia, których nie ma język angielski (...) przykład: język polski dokładnie rozróżnia stryja od wuja, siostrzenicę od bratanicy, czego nie ma język angielski, a przecież struktura rodziny współczesnej jest wszędzie taka sama, mogłoby się więc wydawać, że wyodrębnienie nazwowe stopni pokrewieństwa i powinowactwa rodzinnego powinno być we wszystkich językach takie samo. A nie jest (...). Wynika to z uwarunkowań historycznych i społecznych".

Z wypowiedzi tej wynika oczywisty wniosek, że tłumacząc jeden język na drugi nie można czynić tego mechanicznie, bez uwzględniania specyfiki każdego z tych języków.

A więc i w informatyce tłumaczeń z angielskiego na polski nie należy dokonywać w sposób toporny, sposobem „na wprost"; muszą być uwzględnione wszystkie subtelności językowe.

To jedno. A drugie — to kwestia polskości języka informatyków w ogóle. Nie jest bowiem słuszne ani stosowanie w czystej formie słownictwa angielskiego w odniesieniu do rzeczy lub zjawisk, dla których brak jest jeszcze słownictwa polskiego, ani tym bardziej „spolszczanie" wyrazów angielskich.

Nasz rodak, od ponad 30 lat zamieszkały w USA Roman J. Skonieczny m. in. tak pisze do „Przekroju" [10]: „...większą część swego życia spędziłem wśród ludzi nie mówiących moim ojczystym językiem. Rozmawiając z rodakami za granicą, co w pewnych okresach zdarzało się rzadko, starałem się mówić po polsku, nie wtrącając słów angielskich. Od 1967 roku byłem w Polsce osiem razy (...) zauważyłem jeden niemiły fakt: moi kochani rodacy i ich prasa (...) wprowadzają do naszej bogatej, pięknej mowy obce słowa, szczególnie angielskie. Dlaczego? Czy co obce, to lepsze? (...) Doskonale rozumiem, że słownictwo w wielu dziedzinach współczesnego życia rozwija się z roku na rok i trudno czasem znaleźć odpowiednik w języku polskim (...), ale skoro już mamy polskie, własne słowa — nie zastępujmy ich obcymi".

A zatem, jeśli wszyscy interesujący się kwestią językową w informatyce przejmą się treścią i duchem przytoczonych wypowiedzi, wówczas język informatyków będzie językiem szeroko zrozumiałym i urzeczywistni funkcje, jakie przypisuje się informatyce do spełnienia tak w sferze nauki, jak i praktyki.

W innym bowiem przypadku wyniknie sytuacja, kiedy to — trawestując słynne powiedzenie Mikołaja Reja — będzie można powiedzieć, że informatycy nie ojczysty, a swój — indyczy język mają Gul, gul, gul!!!

Włodzimierz Rabij

LITERATURA

[1] O mowo polska! — Red. K. Nastulanka, Polityka nr 23 (1005) z 5.06.1976 r.
[2] vide — m. in. spec. cykle artykułów w Expresie Wieczornym i Polityce
[3] E. WIECZ — Co to jest nogawka? — Express Wieczorny nr 154 z 10/11.07.1976 r.
[4] T. Rawiński — Kilka uwag na temat słownictwa informatycznego — INFORMATYKA nr 9/1975 r.
[5] D. Kołacka i in. — Terminologia — INFORMATYKA nr 1/1976 r.
[6] Pr. zb. pod red. nauk. Z. HELLWIGA — Automatyczne przetwarzanie informacji — wyd. PWE 1976 r. wyd. II popr. i uzupełn. (szczególnie rozdz. 3 i 4, bo następne w kwestiach językowych są lepsze)
[7] Zob. TEZAURUS z zakresu informatyki — Wprowadzenie — wyd. OBRI Warszawa 1976 r.
[8] Zob. G. EMERY — Elektroniczne przetwarzanie informacji w tłum. M. M. Rachowicz i J. Olszewskiego — wyd. WNT 1975 r. str. 11
[9] W. Rabij — Ułatwiać a nie udziwiać, INFORMATYKA nr 4/1976 r. i J. Kozak — Uwagi i propozycje do terminologii struktur danych — INFORMATYKA nr 6/1976 r.
[10] Przekrój — nr 1625 z 1976 r. str. 14, Po co obce słowa.

O TEZAURUSIE

Zgodnie z zapowiedzią (INFORMATYKA nr 1/1976) ukazał się już Tezaurus z zakresu informatyki. Został on opracowany w Działowym Ośrodku Informacji Ośrodka Badawczo-Rozwojowego Informatyki przez zespół pod kierownictwem mgr Stanisława Zadrożnego. Ma służyć wielu systemom informacyjno-wyszukiwawczym, jakie powstaną w sieci ZOI-ZETO.

W tradycyjnie zorganizowanych ośrodkach informacji naukowej, technicznej i ekonomicznej sposób sporządzania opisu dokumentacyjnego poszczególnych pozycji bibliograficznych pozwalał na pewną dowolność w doborze słownictwa. Ponieważ nie została opracowana jednolita terminologia, wybór słów kluczowych (jedno lub wielowyrazowych nazw przedmiotów, czynności, pojęć etc. zaczerpniętych z języka naturalnego charakteryzujących dany tekst i służących do jego indeksacji) był kwestią indywidualnego, subiektywnego odczucia pracownika dokumentalisty. Zdarzało się, że te same pozycje w kilku ośrodkach INTE były w różny sposób opracowywane, co utrudniało wymianę informacji. Podobna sytuacja istniała w sieci ZETO Zjednoczenia Informatyki.

Efektywne wprowadzenie techniki komputerowej wymaga opracowania tezaurusa; słownika zawierającego terminy z określonej dziedziny zwane deskryptorami, zalecane do opisu treści dokumentu. W miarę potrzeby może on zawierać również terminy niezalecane do stosowania w tym celu (askryptory) z zaznaczeniem ich powiązania z zalecanymi (wg PN-74/N-01180/00).

Dla potrzeb konkretnego systemu i konkretnych zbiorów na podstawie tezaurusa będzie tworzony język informacyjno-wyszukiwawczy o podobnej strukturze i obszarze zawężonym ściśle do tematyki zbioru. Polega to na indeksowaniu poszczególnych pozycji zbioru za pomocą deskryptorów, modyfikatorów i indentyfikatorów zaczerpniętych z tezaurusa. Początkowo szybko rosnący indeks po pewnym czasie ustala swą objętość i wykształca się w język informacyjno-wyszukiwawczy.

Przy opracowaniu tezaurusa w przeważającej części terminologia została zaczerpnięta z opisów dokumentacyjnych zbiorów bibliograficznych i faktograficznych (np. zawierających parametry techniczne) wiodących placówek ZETO, toteż należy oczekiwać, że okaże się on bardzo pomocny w pracy ośrodków.

Tezaurus składa się z 2 części: alfabetycznej — zawierającej cztery wersje językowe: angielską, francuską, niemiecką i rosyjską oraz systematycznej.

W części alfabetycznej poza powszechnie stosowanymi relacjami deskryptorowymi wprowadzono dodatkowo dwie relacje zwrotne: deskryptor działowy oznaczony symbolem DZ i deskryptor grupowy oznaczony symbolem DG.

W części systematycznej składającej się z trzech działów (Terminy podstawowe, Terminy pomocnicze, Terminy uzupełniające) zastosowano zasadę grupowania aspektowego. Oznaczenia haseł za pomocą przestępującego dwuczłonowego numeru pozwalają w razie potrzeby na dodawanie haseł w obrębie poszczególnych grup tematycznych oraz na wprowadzenie nowych grup bez zakłócenia logicznej struktury.

W następnym wydaniu przewiduje się między innymi opracowanie oznaczeń kropkowych ukazujących zależności w obrębie podgrup oraz wykresów graficznych dla zobrazowania związków między deskryptorami niezależnie od usytuowania w poszczególnych grupach.

Tezaurus w obecnej formie powinien okazać się wystarczający dla potrzeb ośrodków INTE Zjednoczenia Informatyki przez najbliższe dwa lata. Niezależnie od tego doraźną aktualizację ma zapewnić coroczny aneks zmian i uzupełnień.

Krystyna SIWEK

*) „Tezaurus z zakresu informatyki". OBRI Warszawa 1976 r.

JERZY KURKIEWICZ
Zjednoczenie Informatyki
KRZYSZTOF ODOLAK
WOJCIECH ZOŁOTUCHO
Instytut Łączności
Warszawa

Sieci transmisji danych—problematyka organizacji i funkcjonowania

Stale rosnące zapotrzebowanie na usługi informatyczne znajduje w ostatnich latach wyraz w coraz większej ilości oddawanych do eksploatacji systemów wielodostępnych i wielokomputerowych. Jednocześnie bardzo szybko rośnie liczba „zdalnych" użytkowników takich systemów. Nowoczesne sieci transmisji danych stwarzają techniczne warunki do funkcjonowania systemów wielodostępnych oraz do budowy sieci i systemów wielokomputerowych.

Prowadzone w szeregu krajów prace teoretyczne i doświadczalne doprowadziły do budowy eksperymentalnych i użytkowych sieci transmisji danych. Prace nad tymi problemami w Polsce prowadzone są między innymi w Instytucie Łączności, Politechnice Gdańskiej, Warszawskiej i Wrocławskiej.

Zagadnienia normalizacji i standaryzacji w odniesieniu do sieci transmisji danych są natomiast przedmiotem studiów Międzynarodowego Komitetu Doradczego do spraw Telefonii i Telegrafii (CCITT) oraz Międzynarodowej Organizacji Normalizacyjnej (ISO).

Plany rozwoju sieci transmisji danych i przegląd dotychczasowych osiągnięć zostały zaprezentowane w czerwcu 1975 roku w Gdańsku na Krajowej Konferencji Naukowo-technicznej „Sieci Teleinformatyczne". W celu przybliżenia tej problematyki szerszym kręgom, proponujemy czytelnikom INFORMATYKI cykl poświęcony sieciom transmisji danych. Materiał ujęty w tym cyklu wybiega poza dotychczas spotykane w kraju rozwiązania techniczne i systemowe.

REDAKCJA

Rozwój systemów przestrzennie rozległych, zwanych w dalszej części cyklu systemami teleinformatycznymi, spowodował powstanie wyspecjalizowanych podsystemów telekomunikacyjnych.

Istotą systemu teleinformatycznego jest możliwość wymiany informacji między zdalnymi użytkownikami. Elementem służącym do przekazywania tych informacji jest podsystem telekomunikacyjny stanowiący integralną część systemu teleinformatycznego.

Strukturę, w skład której wchodzą systemy teleinformatyczne połączone podsystemem telekomunikacyjnym, nazywać będziemy w dalszej części cyklu **siecią teleinformatyczną.**

Natomiast podsystem telekomunikacyjny, który jest pojęciem węższym będziemy utożsamiać z **siecią transmisji danych,** która stwarza techniczne i funkcjonalne warunki do przesyłania informacji cyfrowych pomiędzy abonentami systemów teleinformatycznych.

W niniejszym cyklu poruszone zostaną zagadnienia związane z organizacją i funkcjonowaniem nowoczesnych sieci transmisji danych. Autorzy cyklu zakładają znajomość przez czytelnika podstawowych zagadnień i definicji zakresu transmisji danych, której dobrym uzupełnieniem jest książka James'a Martina pt. „Wprowadzenie do transmisji danych" wydana przez WNT w 1975 r.

W kolejnych artykułach cyklu omówione zostaną dwie podstawowe techniki komutacji stosowane w sieciach transmisji danych:
— komutacja pakietów
— komutacja kanałów.

Technika komutacji pakietów polega na przesyłaniu przez sieć informacji zawartej w standardowych blokach (tzw. pakietach) o określonej maksymalnej długości.

Każdy z pakietów zawiera nagłówek sieciowy z niezbędnymi danymi potrzebnymi do prawidłowego przesyłania (adresy, typ informacji itp.). Pakiet jest niezależnie przesyłany do miejsca przeznaczenia (oznacza to, że dwa kolejne pakiety tej samej informacji mogą być przesyłane do miejsca przeznaczenia różnymi drogami).

Technika komutacji kanałów polega na fizycznym zestawieniu połączenia między dwoma abonentami. W dotychczasowych rozwiązaniach komutacja kanałów znalazła

Mgr inż. JERZY KURKIEWICZ jest absolwentem Politechniki Warszawskiej (1970). W 1971 r. pracował w Rejonowym Centrum Obliczeniowym przy Zakładach Wytwórczych Wysokiego Napięcia. W latach 1973—1974 pracował w Instytucie Łączności w Pracowni Zastosowań Systemów Teleinformatycznych Zakładu Transmisji Danych. Od r. 1974 jest projektantem systemów w Wydziale Techniki i Systemów Komputerowych Zjednoczenia Informatyki.

Mgr inż. KRZYSZTOF ODOLAK adiunkt w Instytucie Łączności jest absolwentem Politechniki Warszawskiej (1970). Od 1970 pracował w Zakładzie Transmisji Danych Instytutu Systemów Telekomunikacji Wojskowej Akademii Technicznej. W 1973 r. przenosi się do Instytutu Łączności, gdzie kieruje kolejno grupami problemowymi ds. Sprzętu i Oprogramowania Telekomunikacyjnego oraz ds. Węzłów Komutacyjnych zajmując się zagadnieniami sieci teleinformatycznych. Brał czynny udział w pracach projektowych nad Eksperymentalną Magistralą Teleinformatyczną Gdańsk — Warszawa — Katowice.

mniejsze zastosowanie ze względu na fakt, że wykorzystywano do tych celów istniejące sieci telefoniczne z komutacją przestrzenną.
Sieci takie charakteryzowały się długim czasem zestawienia połączenia, wysoką stopą błędów oraz nieprzewidzianymi przerwami w transmisji. Natomiast nowoczesna elektroniczna komutacja kanałów czasowych (np. w oparciu o technikę PCM [1]) stwarza perspektywy budowy sieci transmisji danych o zdecydowanie lepszych parametrach technicznych i może być alternatywnym rozwiązaniem w stosunku do sieci z komutacją pakietów. Możliwa jest również budowa sieci wykorzystujących obydwie techniki komutacyjne.

Z punktu widzenia abonenta sieci transmisji danych istotnym zagadnieniem jest rozwiązanie interfejsu i sposobu współpracy pomiędzy urządzeniem końcowym transmisji danych (ang. DTE — Data Terminal Equipment) a urządzeniem komunikacyjnym transmisji danych nazywanym również przyłączem sieciowym (ang. DCE — Data Circuit Terminating Equipment).

Współpraca w relacji abonent — sieć określona jest przez zastosowaną metodę sygnalizacji w interfejsie DTE-DCE. Prace poświęcone zagadnieniom sieci transmisji danych prowadzone są w wielu placówkach badawczych i uczelniach w kraju.

Szczególnie istotnym zagadnieniem jest standaryzacja w skali międzynarodowej podstawowych parametrów i sposobu działania tych sieci. Współczesna sieć transmisji danych powinna charakteryzować się następującymi cechami:

- zminimalizowaniem kosztów jakie będzie ponosił użytkownik sieci
- krótkimi czasami nawiązywania i rozłączania połączeń oraz zminimalizowanie średniego czasu przesyłania informacji
- zapewnieniem obsługi użytkowników różnych klas, klasy użytkowników określają stosowane szybkości, sposób transmisji i rodzaj stosowanego kodu
- zmniejszeniem stopy błędów pierwotnych w porównaniu ze stopą błędów w obecnych sieciach telekomunikacyjnych
- zapewnieniem szerokiej skali ułatwień użytkowych proponowanych dla abonentów sieci,

a w szczególności:

— przezroczystości kodowej
— konwersji szybkości i kodów
— możliwości tworzenia specjalnych wydzielonych sieci z zapewnieniem dostępu przez zastosowanie odpowiedniego klucza
— różnych metod adresowania (adresowanie bezpośrednie, skrócone, wieloadresowanie)
— zwiększenie niezawodności połączeń oraz niezawodności sieci jako całości
— możliwości współpracy z innymi krajowymi i międzynarodowymi sieciami.

Spełnienie wszystkich wymienionych powyżej wymagań jest niemożliwe przy wykorzystaniu obecnie istniejących sieci telekomunikacyjnych.

Dlatego w przyszłości przewiduje się utworzenie specjalnej sieci transmisji danych powszechnego użytku.

[1]) ang. Pulse Code Modulation czyli modulacja kodowo-impulsowa.

Mgr inż. WOJCIECH ZOŁOTUCHO adiunkt w Instytucie Łączności w Zakładzie Teleinformatyki jest absolwentem Politechniki Warszawskiej (1970). W 1970 r. podejmuje pracę w Ośrodku Zautomatyzowanych Obliczeń PKP. Od 1972 r. pracuje w Instytucie Łączności, gdzie zajmuje się zagadnieniami sieci teleinformatycznych. Brał czynny udział w pracach projektowych nad Eksperymentalną Magistralą Teleinformatyczną, kierował grupą problemową ds. „Struktur sieci teleinformatycznych".

Realizacja ww. wymagań może być osiągnięta przez zastosowanie nowoczesnych metod komutacji oraz zastosowanie technik zwielokrotnienia i koncentracji. Dzięki zastosowaniu komutacji elektronicznej uzyskanie krótkiego czasu tworzenia połączeń nie przedstawia zbytnich trudności. Studia nad zagadnieniem powszechnych sieci transmisji danych prowadzone są przez Międzynarodowy Komitet Doradczy ds. Telefonii i Telegrafii (CCITT) oraz przez Międzynarodową Organizację Normalizacyjną (ISO).

Większość prac jest finalizowana na terenie CCITT. Polska jest również członkiem CCITT i bierze udział w pracach tej organizacji.

W roku 1968 CCITT powołało Grupę Mieszaną NRD do pracy nad zagadnieniem nowych sieci transmisji danych. GM NRD w okresie swej pracy wyłoniła 3 grupy robocze:
— Grupa Robocza NRD1 — Współpraca między sieciami transmisji danych
— Grupa Robocza NRD2 — Interfejs między urządzeniami stacji końcowych i siecią transmisji danych
— Grupa Robocza NRD3 — Potrzeby i wymagania użytkowe stawiane sieciom transmisji danych.

Przy czym grupa robocza NRD1 powołała jeszcze dwie podgrupy robocze:
— NRD1/1 — do spraw sieci synchronicznych
— NRD2/2 — do spraw sieci asynchronicznych.

Pomimo krótkiego okresu działalności GM NRD opracowała sześć nowych zaleceń dotyczących przyszłych sieci transmisji danych powszechnego użytku, które dały początek nowej serii o symbolu X. Są to zalecenia X1, X2, X20, X21, X50, X70.

Na V Zebraniu Plenarnym CCITT (1972) postanowiono GM NRD przekształcić w Stałą Komisję Studiów VII. Program studiów tej komisji jest obszerny i obejmuje zagadnienia przedstawione w tabeli.

Tabela 1

Punkt	Temat
A	Standaryzacja kategorii użytkowników sieci transmisji danych powszechnego użytku
B	Standaryzacja udogodnień oferowanych użytkownikom
C	Komutacja pakietów
D	Czasy zestawienia i rozłączenia połączeń
E	Plan numeracji dla sieci transmisji danych powszechnego użytku
F	Jakość załatwiania ruchu
G	Jakość transmisji
H	Potencjalne problemy dla usług teleksowych
I	Standaryzacja sygnalizacji w arytmicznych sieciach transmisji danych powszechnego użytku
J	Wprowadzenie udogodnień dla użytkowników w sieciach transmisji danych powszechnego użytku
K	Wprowadzenie do interfejsu między DTE a DCE możliwości identyfikacji i izolacji usterek
L	Zastosowanie w sieciach transmisji danych powszechnego użytku urządzeń transmisji danych przystosowanych do współpracy z istniejącymi modelami serii V
M	Protekcja znaków kontrolnych używanych do sygnalizacji w interfejsie DCE—DTE
N	Sposób pracy, gdy DTE jest w stanie lokalnym lub nieczynnym
O	Kompatybilność między interfejsami DCE—DTE używanymi do pracy arytmicznej i synchronicznej
P	Interfejs między DTE a DCE dla pracy arytmicznej
Q	Interfejs między DTE a DCE dla pracy synchronicznej
R	Definicje terminów powstających w trakcie studiów nad sieciami transmisji danych powszechnego użytku
S	Zwielokrotniona struktura międzynarodowych łączy między synchronicznymi sieciami transmisji danych
T	Rytm i synchronizacja
U	Zcentralizowana sygnalizacja sterująca dla międzynarodowych połączeń synchronicznymi sieciami transmisji danych
W	Własności i wymagania w stosunku do kanałów podkładanych do przenoszenia niejednokrotnie zwielokratnianych kanałów o łącznej przeplywości binarnej 64 kbit/s
X	Kontrola błędów
Y	Zabezpieczenie w sieciach transmisji danych

Szeroki zakres studiowanych zagadnień odzwierciedla problemy nowoczesnych sieci transmisji danych.

W chwili obecnej wynikiem prac tej komisji jest szereg nowych zaleceń serii X oraz modyfikacja poprzednio wydanych.

Do chwili obecnej opracowano (lub są w trakcie opracowywania) następujące zalecenia:

Usługi i ułatwienia użytkowe

X1 Międzynarodowe kategorie usług w powszechnych sieciach transmisji danych

X2 Międzynarodowe ułatwienia użytkowe w powszechnych sieciach transmisji danych.

Interfejs i wyposażenie terminala

X20 Interfejs pomiędzy DTE i DCE dla transmisji start-stopowej w powszechnych sieciach transmisji danych

X20 bis Interfejs kompatybilny z interfejsem V21 pomiędzy DTE i DCE dla transmisji start-stopowej w sieciach transmisji danych

X21 Interfejs pomiędzy DTE i DCE dla synchronicznych klas użytkowników w powszechnych sieciach transmisji danych

X21 bis Stosowanie w powszechnych sieciach transmisji danych DTE zaprojektowanego dla interfejsu z modemami serii V

X24 Lista definicji dla obwodów pomiędzy DTE i DCE w powszechnych sieciach transmisji danych

X25 Interfejs pomiędzy DTE i DCE dla terminali pracujących pakietowo w powszechnych sieciach transmisji danych

X30 Standaryzacja podstawowego modelu drukarki zgodnie z Międzynarodowym Alfabetem nr 5

X31 Charakterystyki z punktu widzenia transmisji pomiędzy DTE i DCE stosującymi transmisję start-stopową 200 bodów i Międzynarodowy Alfabet nr 5

X32 Jednostki odpowiedzi zwrotnej dla urządzeń start-stopowych 200 bodowych zgodnie z Międzynarodowym Alfabetem nr 5

X33 Standaryzacja międzynarodowego tekstu dla pomiarów marży urządzeń start-stopowych zgodnie z Alfabetem nr 5

Transmisja, sygnalizacja i komutacja

X40 Standaryzacja systemów transmisyjnych ze skokową modulacją częstotliwości dla zapewnienia telegraficznych i telefonicznych kanałów przez podział częstotliwości grupy pierwotnej

X50 Podstawowe parametry planu zwielokrotnienia dla międzynarodowego interfejsu pomiędzy synchronicznymi sieciami transmisji danych

X51 Podstawowe parametry planu zwielokrotnienia dla międzynarodowego interfejsu pomiędzy synchronicznymi sieciami transmisji danych stosującymi 10-bitową strukturę koperty

X60 Sygnalizacja we wspólnym kanale dla synchronicznej transmisji danych — część dotycząca użytkownika

X70 Terminalowy i tranzytowy system sygnalizacji dla usług start-stopowych dla międzynarodowych linii pomiędzy asynchronicznymi sieciami transmisji danych

X71 Zdecentralizowany system terminalowej i tranzytowej sygnalizacji dla międzynarodowych linii pomiędzy synchronicznymi sieciami transmisji danych.

Parametry sieci

X92 Hipotetyczne połączenie odniesienia dla synchronicznych powszechnych sieci transmisji danych

X93 Hipotetyczne połączenie odniesienia dla usługi komutacji pakietów

X95 Sieciowe parametry w powszechnych sieciach transmisji danych

X96 Sygnały progresji wywołania w powszechnych sieciach transmisji danych.

Niektóre z przytoczonych tutaj zaleceń zostaną omówione w kolejnych artykułach tego cyklu.

LITERATURA

[1] James Martin — Wprowadzenie do transmisji danych, WNT 1975 r.

[2] L. Kwiatkowski, K. Odolak — Podstawowe parametry techniczne sieci teleinformatycznych jako czynnik determinujący charakter i jakość świadczonych usług. Konferencja Naukowo-Techniczna Sieci Teleinformatyczne, Gdańsk 1975

[3] J. Kurkiewicz, S. Michalak — Przegląd istniejących i projektowanych sieci teleinformatycznych. Konferencja Naukowo-Techniczna. Sieci Teleinformatyczne, Gdańsk, 1975

[4] Materiały Komisji VII CCITT.

NASZE RECENZJE

Pomiędzy człowiekiem a maszyną cyfrową

Naukowcy, którzy od społecznej i psychologicznej strony obserwują rozwój naszej cywilizacji, coraz bardziej zdecydowanie stwierdzają, że człowiek przestaje nadążać za zmianami otaczającego go świata. Pojawiające się wciąż nowe techniczne wynalazki stwarzają całkiem odmienne warunki egzystencji, zmuszając ludzi do rezygnacji z wielu dotychczasowych przyzwyczajeń, czasem nawet do częściowych zmian osobowości. Ludzie zaś na takie metaforfozy nie mają ochoty, albo po prostu nie potrafią się dostatecznie szybko zmienić. Są raczej zaszokowani ekspansją świata techniki we własne prywatne dotąd sprawy (stąd też najbardziej znana książka poruszająca ten temat nosi nazwę „Szok przyszłości" napisał ją amerykański futurolog Alvin Toffler). Za szczególnie oczywisty przykład tej ekspansji, za urządzenie pojawiające się jakby „z przyszłości" i nie pasujące do obecnych możliwości ludzkiej adaptacji, uznany został właśnie komputer. Maszyny cyfrowe obciąża się odpowiedzialnością za wymuszanie cybernetycznych metod kierowania społeczeństwami, uważa się je za urządzenia tajemnicze i niedostępne dla przeciętnego człowieka, które jednak bezceremonialnie ingerują w jego życie doprowadzając do rozmaitych psychicznych napięć.

W tej sytuacji konieczne stało się zwrócenie większej uwagi na zagadnienia pojawiające się na styku maszyna-człowiek. Na rozpatrzenie ich nie tylko z punktu widzenia technologii środków porozumiewania się operatora z komputerem, ale także od strony łatwości nawiązywania kontaktu, komunikatywności, ilości i różnorodności wzajemnej informacji oraz psychologicznych aspektów naszego dialogu z komputerem. Owa potrzeba spojrzenia na problem z wielu stron powoduje, iż staje się on bardzo szeroki i trudny do ujęcia. Nic więc dziwnego, że chcąc omówić ten pro-

*) James Martin — Dialog człowieka z maszyną cyfrową, tłumaczyli: Cz. Syc, L. Maciejec, T. Sinkiewicz, A. Zakrzewski; Wydawnictwa Naukowo-Techniczne, seria „Informatyka", Warszawa 1976, stron 537, cena 98 zł.

blem bez przesadnych uproszczeń, trzeba poświęcić mu ponad pół tysiąca stron dużego formatu. Nawet wówczas, jeśli jest się tak rutynowanym popularyzatorem i autorem podręczników jak James Martin.

Próbkę złożoności zagadnienia otrzymujemy już w traktowanej jako wprowadzenie pierwszej części książki. Wbrew pozorom nie rozmaitość urządzeń technicznych i możliwości ich wykorzystania jest tu największym kłopotem — zagadnieniom sprzętu poświęcony jest jedynie skromny podrozdział. Znacznie obszerniej prezentowane są rodzaje i struktura dialogów oraz ocena człowieka, który má w nich uczestniczyć. Przy tej okazji wypada uznać odwagę Martina, który wprowadza pojęcie operatora inteligentnego, nietolerancyjnego, czy też bardzo niecierpliwego. Być może zgodnie z zasadami dobrego wychowania, nie wypada traktować operatora inaczej niż jako bystrego i wyrozumiałego. Pamiętajmy jednak, że coraz powszedniejsze użycie komputerów zmusza do kontaktów z nimi rozmaitych ludzi. A więc takich, których np. współczynnik inteligencji jest niższy od podobnego współczynnika informatyków budujących maszyny i tworzących systemy konwersacyjne. Nadmierna skromność tych ostatnich, przecenianie zdolności użytkownika doprowadza wówczas do powstawania systemów konwersacyjnych zbyt trudnych dla przeciętnego operatora.

Druga część książki przeznaczona została na przedstawienie najczęściej stosowanej formy kontaktu człowieka z maszyną to jest dialogów alfanumerycznych. W osobnych rozdziałach znaleźć można podstawowe wiadomości o dialogach przy użyciu konwersacyjnych języków programowania, za pomocą pióra świetlnego i alfanumerycznych monitorów ekranowych (autor wymienia dwadzieścia trzy metody dialogu poprzez monitor), a także omówienie czynności sterujących dialogiem poprzez urządzenia ekranowe. Najbardziej atrakcyjne są oczywiście dialogi w języku naturalnym (też odrębny rozdział). Za przykład służy tu system ELIZA opracowany w Massachussetts Institute of Technology w ramach projektu MAC. Ponieważ Martin tego nie podaje wyjaśnię, że MAC czyli Multi Access Consultant (Wielodostępny Konsultant) jest projektem mającym wspomagać lekarzy w czasie przeprowadzania wywiadu z pacjentami, w rozpoznawaniu chorób i przy ustalaniu najwłaściwszej diagnozy. ELIZA, działająca w ramach MAC już od dziesięciu lat opracowuje również wywiady z niezrównoważonymi psychicznie pacjentami. Znacznie ciekawszy od dialogu podanego w tekście z „przygnębioną dziewczyną" jest inny przykład działania ELIZY, który został umieszczony w dodatku na końcu książki. Jest to mianowicie rozmowa jaką przeprowadził z ELIZĄ lekarz psychiatra. Posługując się dalekopisem nie wiedział on, że porozumiewa się z maszyną. Sądząc, iż ma do czynienia z kolegą po fachu, zwierzał się komputerowi ze swoich zastrzeżeń wobec maszynowego badania pacjentów: w MIT istnieje program maszynowy, który nas zrujnuje... w programie maszyny nie ma nic twórczego... maszyny bawią się liczbami, to wszystko". Po krótkiej wymianie zdań wystraszony psychiatra znajduje bratnią duszę i pocieszyciela: „Żadna maszyna nie potrafiłaby rozmawiać tak jak pan... pan jest dla mnie podporą". Obaj mają podobny sposób rozumowania, co zdaniem psychiatry niezbicie „pokazuje różnicę między człowiekiem a maszyną". „Rozumiem" odpowiada mu komputer.

Jak to jest możliwe? Przecież każdy język naturalny jest wystarczająco zagmatwany, wieloznaczny i bogaty w wyjątki, żeby przyprawić najmądrzejszy komputer o zawrót głowy. I tak jest istotnie! Maszyna nie rozumie bynajmniej odpowiedzi swojego rozmówcy — wybiera tylko z niej jakieś znane sobie słowo i traktuje je jako podstawę do zbudowania własnego pytania. Tak więc komputer przy prowadzeniu dialogu z człowiekiem nie kieruje się jakimś ustalonym, konsekwentnym przewodem logicznym. Działa od odpowiedzi do odpowiedzi, szukając za każdym razem słów — haczyków, o które można się zaczepić tworząc następne pytanie na tyle ogólne, aby mogło wyglądać na sensowną kontynuację rozmowy. Gdy zaś nic nie znajduje, ratuje się powtórzeniami słów człowieka w nieco zmienionej formie (np. pytania, negacji) albo bezpiecznymi zwrotami typu „doprawdy tak pan sądzi?, rozumiem, albo po prostu mówi: och!

W trzeciej części książki James Martin zajmuje się dialogami prowadzonymi za pośrednictwem głosu i obrazu. Urządzenia do tego rodzaju konwersacji nie są jeszcze szeroko rozpowszechnione. Należy się jednak spodziewać, że staną się w przyszłości bardziej popularne ze względu na przystępność takiego dialogu dla nieobeznanego z maszyną użytkownika.

Jest to szczególnie dogodne — np. końcówki ustawione w prywatnych mieszkaniach i podłączone do kabla telewizyjnego mogą w przyszłości pełnić rolę sprzedawcy zachwalając rozmaite potrzebne nam towary wyświetlając jednocześnie ich zdjęcia i targując się o ceny. Obecnie jednak urządzenia obrazowe w praktyce stosowane są prawie wyłącznie do automatyzacji projektowania. Jeszcze mniejsze pole do popisu mają systemy konwersacji fonicznej, choć przecież aparat telefoniczny jest najpowszechniejszym środkiem zdalnej komunikacji między ludźmi. Kolejna część „Dialogu..." poświęcona została na rozważania natury psychologicznej. Można w niej znaleźć sporo ciekawych wiadomości o psychologii użytkownika (np. przyczynach powstawania błędów, które można ograniczyć, gdy „operator ma motywację wykonywanej pracy i jest nią zainteresowany" — niby oczywiste, ale często się o tym zapomina), o znaczeniu czasu reakcji człowieka i jego twórczych możliwości. Rozważania te są kontynuowane również w następnej części omawiającej specyfikę współpracy z operatorami bez przeszkolenia. Znalazł się tutaj także obszerny rozdział na temat nauczania wspomaganego przez maszynę cyfrową — dyscypliny osiągającej ostatnio interesujące rezultaty.

Wreszcie ostatnia, szósta część książki przeznaczona jest na uwagi dotyczące problemów realizacji dialogu. Dużo miejsca zajmują w nim wnioski dotyczące powstawania, wykrywania i usuwania błędów. Jest to bardzo uzasadnione. W różnego rodzaju sprzęcie informatyki, wraz ze wzrostem jego złożoności i gęstości upakowania, coraz większego znaczenia nabierają zagadnienia niezawodności, rozmaite aspekty testowania i diagnostyki. Mniej istotny dla polskiego czytelnika jest natomiast rozdział o ochronie tajności i poufności danych.

Nie ma u nas konkurencyjnych firm wydzierających sobie tajemnice handlowe, a zresztą chcąc uzyskać rzetelną wiedzę na ten temat lepiej, zamiast studiować Martina, przeczytać książkę M. Wessela „Komputer i społeczeństwo", która ukazała się w tym roku w serii „Omega". Także i inne części książki Jamesa Martina nie zawsze przystają do naszych potrzeb. Przykładem jest nadmiernie rozbudowany opis konwersacyjnych systemów handlowych. Nie powinniśmy również ufać prostocie prezentowanych systemów konwersacyjnych — angielski znakomicie nadaje się do tego celu, stworzenie systemów (zwłaszcza fonicznych) działających w języku polskim byłoby dużo trudniejsze.

Nie są to jednak zastrzeżenia istotne. „Dialog człowieka z maszyną cyfrową" czyta się znakomicie. Rysunki, zdjęcia, liczne przykłady (np. dialog kosmonautów z komputerem na filmie „2001: Odyseja kosmiczna"), a przede wszystkim prosty, komunikatywny język powodują, że książka może zainteresować nawet niefachowców.

Nie sposób nie doceniać znaczenia tego typu publikacji. Rozwój informatyki prowadzić będzie przecież do coraz częstszych związków coraz większej liczby ludzi z bardziej złożonymi systemami komputerowymi. Zdanie się w tych kontaktach na żywioł, aranżowanie ich po amatorsku jest na dłuższą metę nie do pomyślenia. Trzeba zatem rozpatrzyć w rozmaitych aspektach wszystkie występujące tutaj zależności, zdać sobie sprawę z tego pod jakimi względami człowiek góruje nad maszyną, a pod jakimi maszyna przewyższa człowieka.

Tylko uświadomienie sobie tych przewag i słabości prowadzić może do w pełni efektywnego wykorzystania ogromnego potencjału maszyny. Do sytuacji, w której człowiek stanie się używając określenia Jamesa Martina „ogonem, który kręci psem".

Marek Hołyński

Współczesne kryteria oceny drukarek

Spośród wszystkich urządzeń peryferyjnych drukarka jest niewątpliwie modułem skupiającym na sobie główną uwagę użytkownika. Wyniki to z faktu, że będąc źródłem ostatecznego produktu przetwarzania w postaci czytelnej dla człowieka stanowi ona łatwo sprawdzalny dowód prawidłowości wykorzystania komputera. Pełnienie tej funkcji powoduje bardzo często, że drukarki należą do najbardziej intensywnie eksploatowanych urządzeń komputerowych, często nawet w sposób rabunkowy, nie zważając na fakt dominującego udziału w nich zespołów typu mechanicznego, wrażliwych na uszkodzenia i ulegających stosunkowo szybkiemu zużyciu. Drukarki komputerowe od prawie 20 lat podlegają nieustannej ewolucji, zmierzającej obok zwiększania szybkości drukowania, również do poprawy niezawodności działania i wskaźników ekonomicznych. W ostatnich latach pojawili się również wyspecjalizowani producenci, którzy oferują coraz szerszy asortyment drukarek o bardzo zróżnicowanych cenach i parametrach eksploatacyjnych. Polska jak wiadomo osiągnęła również znaczne sukcesy w tej dziedzinie dzięki trafnie wybranym licencjom, uzyskując pozycję poważnego eksportera drukarek (MERA-BŁONIE). Utrzymanie tej pozycji w warunkach bardzo ostrej konkurencji na rynku światowym wymaga uważnego śledzenia postępu technicznego.

Obecnie o wyborze drukarki nie decydują już jak przed laty wyłącznie parametry szybkości drukowania, ale cały szereg innych kryteriów, zarówno ekonomicznych (cena zakupu, koszty eksploatacji), jak i technicznych (niezawodność zespołów, łatwość konserwacji, wygoda obsługi, niski poziom hałasu, jakość drukowania, repertuar znaków).

Na czoło tych kryteriów wysuwa się niewątpliwie niezawodność, wpływająca ze względu na funkcje drukarki w całości konfiguracji komputera w decydującym stopniu na poziom całkowitych kosztów eksploatacji.

O niezawodności działania drukarki decyduje z kolei jej podstawowy i najbardziej obciążony pracą zespół, jakim jest mechanizm drukujący. Większość stosowanych obecnie rozwiązań konstrukcyjnych opiera się na elektromagnesie oraz ramieniu dociskowym, które uruchamia młotek drukujący, przy czym młotek ten może mieć stałe lub elastyczne podparcie. Mechanizm młotkowy stosowany np. przez czołowego światowego producenta drukarek, amerykańską firmę DATA PRODUCTS zawiera elastycznie podparty młotek, który jest uruchamiany przez cewkę typu głośnikowego. Sam młotek drukujący jest osadzony na płaskiej cewce i podwieszony na dwóch elastycznych sprężynach, które wykorzystywane są również do doprowadzania prądu wzbudzającego cewkę, umieszczoną między dwoma stałymi magnesami. Wytworzenie przez cewkę pola magnetycznego powoduje szybkie dociśnięcie przez młotek papieru i taśmy barwiącej do czcionki drukowanego znaku. Czas lotu młotka jest bardzo stabilny, co zapewnia uzyskanie wydruku równych wierszy o jakości spełniającej nawet wymagania późniejszego odczytu optycznego dokumentów (OCR). Istnieje możliwość regulacji czasu lotu młotka przez zmianę początkowego jego położenia za pomocą wkrętu oporowego. Spoczynkowe podparcie młotka wykonane jest z podatnego materiału celem tłumienia drgań powstających przy jego powrocie. Regulacja taka jest bardzo łatwa, ponieważ czas lotu młotka można obserwować na ekranie oscyloskopu w postaci odpowiedniego sygnału elektrycznego.

Doświadczenia eksploatacyjne wykazały, że liczba uderzeń pomiędzy uszkodzeniami zespołu uruchamiającego ten młotek wynosi średnio 2 miliardy, natomiast regulacja podparcia spoczynkowego młotka powinna odbywać się co 150 milionów uderzeń. Interesującym faktem jest to, że wspomniana firma DATA PRODUCTS dzięki oparciu się o młotek takiej konstrukcji jako pierwsza osiągnęła szybkość do 2400 uderzeń na minutę i zastosowała go w całej rodzinie drukarek o szybkości od 300 do 2000 wierszy/min.

Pojawienie się minikomputerów spowodowało konieczność poprawy podstawowego wskaźnika efektywności ekonomicznej (stosunek ceny do wydajności urządzenia) w klasie drukarek o średniej szybkości. Cel ten osiągnięto dzięki obniżce kosztów produkcji, m.in. w wyniku zastąpienia w konstrukcji wielu części używanych dotąd metali przez tworzywa sztuczne, a także wprowadzeniu wszechstronnie wypróbowanych w eksploatacji takich podstawowych podzespołów, jak młotki, przesuw papieru za pomocą serwomechanizmu, itp. Np. w nowym modelu 2200 drukarki wierszowej produkcji firmy DATA PRODUCTS zastosowano znacznie ulepszony młotek typu MARK IV b, montowany w poczwórnych modułach i wyposażony w stałe, obudowane w plastyku magnesy, charakteryzujące się o 50% większą gęstością strumienia. Spowodowało to znaczne obniżenie poziomu kosztów eksploatacji w wyniku stosowania mniejszych prądów wzbudzania młotków (mniejsze zużycie energii) oraz prostszych elektronicznych układów sterowania młotków.

Przytoczone fragmentaryczne przykłady ilustrują rodzaj i skalę problemów, jakie obecnie rozwiązują producenci drukarek w dążeniach do poprawy niezawodności i wskaźników eksploatacyjnych drukarek wierszowych.

(WK)

SPROSTOWANIE

W numerze 1/77 INFORMATYKI w artykule **Marii Łąckiej i Jadwigi Tarasiuk pt. Język KWINTET,** wkradł się błąd w podpisach pod rysunkami. Podpisy te powinny brzmieć:

- Rys. 1. Postać zlecenia podstawowego
- Rys. 2. Postać wyrażenia prostego
- Rys. 3. Postać relacji języka
- Rys. 4. Przykład struktury hierarchicznej.

Za błędy przepraszamy Autorki oraz Czytelników

Redakcja

posteor

OFERUJE

PAKIET ORGANIZACJI BIBLIOTEKI W PAMIĘCI BĘBNOWEJ DLA MASZYN CYFROWYCH SERII 1300

- usprawnia organizację procesu obliczeniowego
- umożliwia użytkownikom maszyn serii ODRA 1300 optymalne wykorzystanie pamięci bębnowej w celach operacyjnych
- redukuje znacznie czas pracy ODRY serii 1300
- 4 sekundy – czas odszukania i załadowania średniej wielkości programu
- *koszt wdrożenia 90000 zł*
- *efekt ekonomiczny 4 – 5 mln zł*

WOT/1521/76

POSTEOR – Przedsiębiorstwo Wdrażania i Upowszechniania Postępu Technicznego i Organizacyjnego
ul. Mickiewicza 60 a, 81-866 Sopot
Telefon: 51 4973; Teleks: 051769